MASTER EN SCIENCES MATHÉMATIQUES

Etude des méthodes de type spectral dans la détection de communautés au sein des réseaux

Buyse, Virginie

Award date:
2018

Awarding institution:
Universite de Namur

UNIVERSITE DE NAMUR

Faculté des Sciences

ÉTUDE DES MÉTHODES DE TYPE SPECTRAL DANS LA DÉTECTION DE COMMUNAUTÉS AU SEIN DES RÉSEAUX

Mémoire présenté pour l'obtention

du grade académique de master en finalité spécialisée en perspectives professionnelles des mathématiques appliquées

Virginie BUYSE

Juin 2018

UNIVERSITE DE NAMUR

Faculté des Sciences

ÉTUDE DES MÉTHODES DE TYPE SPECTRAL DANS LA DÉTECTION DE COMMUNAUTÉS AU SEIN DES RÉSEAUX

Promoteur :

Renaud LAMBIOTTE

Mémoire présenté pour l'obtention

du grade académique de master à finalité spécialisée en perspectives professionnelles des mathématiques appliquées

Virginie BUYSE

Juin 2018

Remerciements

Merci à mon promoteur, Renaud Lambiotte, de m'avoir fait découvrir un sujet aussi passionnant que les réseaux par le cours de « Théorie des graphes » et de m'avoir guidée, accompagnée et rassurée tout au long de ce mémoire. Merci également aux lecteurs de celui-ci d'avoir pris le temps d'évaluer ce travail final.

Merci à L.L., ma sœur de cœur, avec qui j'ai travaillé jusqu'à 4h du matin. Sans toi pour me motiver et décompresser, je n'aurais pu tenir.

Merci à N.H. pour avoir corrigé mes fautes d'orthographes et pour avoir relevé que l'on place des arêtes avec Poisson.

Merci finalement à l'équipe du gîte pour m'avoir remonté le moral dans un moment de grand désespoir. Sans vous, j'aurais peut-être tout lâché ce soir-là.

Abstract

La complexité et la taille sans cesse croissante des réseaux présents dans notre société actuelle rendent les communautés très attractives. Celles-ci sont des groupes de sommets définis tels que la concentration en arêtes en leur sein est plus importante qu'entre deux d'entre elles. Leur détection a constitué un sujet vivement étudié ces dernières années car elles permettent notamment de comprendre la structure des réseaux. Par conséquent, les méthodes créées dans ce contexte sont légion. Ce mémoire en développe trois de nature différente et nous mène vers leur point commun qui serait dans les outils spectraux.

Mots clés : réseau - communauté - outils spectraux - modularité - inférence statistique - stabilité

The complexity and the constantly increasing size of the networks in our current society make communities attractive. These are groups of nodes defined such as the concentration in edges is higher within them than between them. Their detection has been a widely studied topic over the past few years because they allow among other things to understand the network's structure. Consequently, there are plenty of methods created in this context. This master thesis develop three of it with different nature and guide us to their common point which should be in the spectral tools.

Key words : network - community - spectral tools - modularity - statistical inference - stability

Table des matières

1 Introduction

Un réseau est un objet mathématique composé de nœuds, appelés sommets, liés entre eux par des arêtes. Cet outil est tellement général qu'une kyrielle d'applications peuvent y être associées. Les réseaux routiers, les connexions entre neurones, les cercles d'amis... tous sont à même d'être modélisés par cet objet. Dans ce mémoire, nous nous intéressons aux communautés des réseaux, ce sont des groupes de sommets définis par une forte concentration d'arêtes en leur sein. La pluridisciplinarité des applications des réseaux, et en particulier de ces communautés, a su créer un vif intérêt chez les chercheurs, développant ainsi une multitude de méthodes quant à la détection de ces dernières. En ce sens, nos objectifs sont la réalisation d'une synthèse de méthodes de nature différente et la recherche d'un point commun entre celles-ci. L'intuition de l'identité de ce dernier, apportée par notre promoteur, nous guide vers les éléments spectraux des matrices représentant les réseaux.

Pour atteindre ces objectifs, nous développerons, dans une première partie, une analyse théorique des méthodes. Chacune d'elles sera formellement définie puis entraînée, le cas échéant, vers une version spectrale d'elle-même. Dans une seconde partie, ces méthodes seront implémentées aux fins de les appliquer sur des réseaux concrets et de vérifier si elles se rejoignent également en pratique. Avant tout ceci, dans un chapitre introductif, quelques indices quant à l'existence des communautés seront présentés ainsi que l'intérêt d'étudier de tels éléments. Quelques notions théoriques seront ensuite rappelées, suivies d'une brève description des méthodes de détection de ces groupes de sommets.

Préambule

Dans le monde qui nous entoure, nous sommes toujours confrontés aux réseaux, que ce soit en ouvrant une page internet ou en se rendant au travail en voiture. Conséquemment, de nombreux systèmes peuvent être représentés par ceux-ci, notamment le réseau routier, les connexions existant au sein du cerveau ainsi que les relations entre collègues dans une entreprise. Être capable d'analyser un tel objet mathématique représente donc un bel atout pour le mathématicien dans notre société actuelle.

Ce chapitre met en lumière le contexte de la problématique de la détection de communautés ainsi que les objectifs de ce mémoire. Ensuite, il énonce plusieurs raisons expliquant la présence de communautés dans un réseau avant de présenter l'intérêt de les rechercher. Il rappelle également les notions théoriques nécessaires à la compréhension du lecteur. Enfin, il introduit les diverses classes de méthodes de détection de communautés qui seront abordées dans les chapitres ultérieurs.

1.1 Contexte et objectifs

La rédaction de cette section se base sur les éléments apportés par [Porter et al., 2009]. Nous allons décrire le contexte dans lequel se déroule l'étude des communautés d'un réseau ainsi que la raison pour laquelle il est intéressant de les découvrir. Ceci nous mènera à la description des objectifs de ce mémoire.

Il est possible d'étudier un réseau à plusieurs niveaux. Premièrement, certains peuvent être intéressés par l'analyse de la structure des interactions et donc des arêtes elles-mêmes. En effet, elles peuvent avoir un poids, donnant plus d'importance à certaines d'entre elles, et une direction. Cela représente l'étude du réseau au niveau microscopique. A contrario, l'étude au niveau macroscopique analyse le comportement de groupes de sommets interagissant ou non entre eux. Ce n'est donc pas la dynamique de chaque particule qui est considérée mais une moyenne de tous les comportements microscopiques. Enfin, il existe un niveau intermédiaire, ou mésoscopique, qui reprend l'étude de structures au sein du réseau qui sont « suffisamment grandes pour pouvoir analyser leurs propriétés collectives mais suffisamment petites pour supposer que celles-ci sont obtenues par la moyennisation des comportements des constituants de ces structures » ([Porter et al., 2009]). Par conséquent, l'étude au niveau mésoscopique est très intéressante lorsque nous traitons des données de taille importante. Les *communautés* forment un exemple d'une telle structure. Ce sont des sous-ensembles de sommets du réseau tels qu'il existe de nombreuses connexions entre ceux d'un même groupe et très peu entre ceux de deux groupes différents.

La détection de communautés a connu un franc succès ces dernières années et diverses méthodes de résolution pour ce problème ont été implémentées à partir de statistiques, de sociologie, de mathématiques discrètes et bien d'autres encore. La résolution de ce problème ne réunit donc pas uniquement des mathématiciens mais également d'autres groupes de chercheurs. Cela s'explique de part la nature interdisciplinaire des communautés. En effet, celles-ci se retrouvent en sociologie où elles peuvent représenter des cercles d'amis dans un réseau social, en génétique où elles peuvent représenter des groupes de cellules associés à des fonctionnalités différentes ou encore sur Internet où elles peuvent représenter des ensembles de pages liées à des sujets différents. Une autre explication de cette interdisciplinarité vient du fait que l'interprétation de la structure des communautés et de leur comportement demande un savoir spécifique au contexte de l'étude.

Dans cet ordre d'idées, « un trait rendant le problème de détection de communautés si difficile est la spécificité de leur définition au domaine d'étude. En deuxième lieu, une fois qu'une bonne formulation adaptée au contexte a été choisie, il reste à résoudre un problème d'optimisation généralement NP-complet » ([Porter et al., 2009]). Ceci implique la nécessité d'adapter des méthodes d'optimisation heuristiques ou d'en créer de nouvelles. Enfin, comme précisé précédemment, ce problème réunit des chercheurs provenant de domaines différents ayant chacun des besoins distincts. Ces trois idées entraînent l'existence d'une grande variété de méthodes de détection de communautés.

Ce mémoire consistera donc à rédiger la synthèse de méthodes de natures différentes et d'en effectuer une comparaison. Plus particulièrement, nous évaluerons la possibilité d'utiliser des outils de types spectraux sur ces dernières, notre intuition étant que ceux-ci constituent le point commun entre ces méthodes.

1.2 Indices quant à l'existence des communautés

En prenant connaissance des objectifs de ce mémoire, une question nous vient à l'esprit : les réseaux sont-ils naturellement divisés en communautés ? Dans cette section, nous énoncerons plusieurs raisons pouvant mener ces derniers à adopter une structure modulaire.

Dans un réseau de relations sociales par exemple, l'existence de communautés est triviale. Nous sommes tous connectés à des amis, à des collègues ou à la famille. Certains liens sont plus forts que d'autres, d'où l'émergence de groupes. Un autre exemple dans lequel l'existence des communautés est aisée à imaginer seraient les réseaux routiers. En effet, les connexions sont plus denses au sein d'une ville qu'entre deux villes.

Afin de savoir ce qui peut mener un réseau à adopter une telle structure, nous nous sommes basés sur [Meunier et al., 2010]. Pour commencer, un des tous premiers arguments mentionnés dans cette source stipule qu'un réseau muni d'une structure modulaire permet une adaptation plus rapide aux changements extérieurs. En effet, ce réseau peut évoluer

module par module sans risquer de perdre certaines parties déjà convenablement adaptées. « Une telle robustesse dans l'évolution représente un atout pour n'importe quel système devant s'adapter ou évoluant par critère de sélection ». Dès lors, cela pourrait expliquer que de nombreux réseaux adoptent la structure de communautés. De plus, dans le même ordre d'idées, un tel réseau peut également être construit plus rapidement et de manière plus robuste.

Suite à cette première idée, de nombreux auteurs se sont penchés sur le sujet, développant ainsi plusieurs arguments susceptibles de justifier l'existence de la structure modulaire des réseaux. Un premier argument apporté sur base du travail de [Sporns et al., 2004] est que les réseaux présentant une structure modulaire pourraient favoriser des traitements d'informations localisés et seraient donc plus efficaces en ce sens. De plus, les chemins entre les éléments d'une même communauté étant courts [1], la transmission d'informations entre ceux-ci n'est pas parasitée par des sources extérieures ou par l'emploi d'intermédiaires pour la réaliser.

En deuxième lieu, la topologie d'une telle structure « produit une séparation entre les échelles temporelles » de la dynamique en un même groupe et de la dynamique entre groupes différents. En effet, comme expliqué dans [Pan & Sinha, 2009], la diffusion se réalise bien plus vite à l'intérieur des communautés qu'entre elles et ce, avec un rapport non-linéaire.

Un argument supplémentaire nous vient de [Robinson et al., 2009] faisant intervenir la stabilité du réseau. Si ce dernier possède une structure de communautés, il peut être divisé tout en le laissant connexe ou sans le rendre instable. En outre, il permet également des combinaisons ou des reconnexions sans trop affecter la stabilité du réseau.

Une autre idée appuyant l'existence de communautés serait le lien entre les mécanismes intervenant dans le développement de réseaux et la formation de celles-ci. En ce sens, un exemple est fourni par [Rubinov et al., 2009]. Le réseau de base est progressivement ajusté en renforçant ou affaiblissant les liens entre des sommets particuliers. Cet ajustement, « similaire à la plasticité neuronale », produit de l'ordre dans les connexions menant à une structure modulaire.

Enfin, un autre argument apparait dans [Kashtan & Alon, 2005] : un réseau composé de communautés est optimal pour évoluer sous un milieu changeant de manière modulaire. Autrement dit, dans une situation où différents objectifs ont des sous-objectifs communs, le réseau va se développer de manière à créer des groupes de sommets chacun spécialisé dans la réalisation d'un sous-objectif particulier.

1. La présence de nombreuses arêtes entre les sommets d'une même communauté raccourci les chemins entre ces derniers.

Il existe encore bien d'autres raisons pouvant expliquer l'existence de communautés dans un réseau mais plus spécifiquement dans un réseau cérébral. Elles sont de nature neuroscientifique, anatomique ou encore psychologique et nous laissons au lecteur intéressé le soin de consulter [Meunier et al., 2010] pour une synthèse de ces raisons.

1.3 Intérêt des communautés

Il nous reste un point supplémentaire à éclaircir avant d'aborder les notions de base nécessaires à la compréhension de ce mémoire : pourquoi les communautés sont-elles intéressantes à détecter ?

« Les chercheurs et scientifiques de tous horizons doivent travailler avec une quantité de données ou d'informations constamment en hausse » ([Reichardt & Bornholdt, 2008]). En raison de cette échelle importante dans la taille des bases de données mais également de la nature assez complexe de nombreux systèmes étudiés, le développement de techniques de regroupement ou de partition est important sous plusieurs points de vue.

D'une part, à grande échelle, l'analyse d'un réseau et de sa structure se complique. Ainsi donc, la division du réseau en communautés pertinentes permettrait de renseigner de manière efficace sur la structure sous-jacente de celui-ci. Pour donner quelques exemples, les regroupements de sommets dans un réseau neuronal permet d'identifier les neurones liés à une même fonctionnalité du cerveau et de ce fait, de comprendre l'organisation interne de ce dernier. Dans un réseau social, les communautés peuvent permettre d'identifier les mécanismes de formation d'amis ou de faire des suppositions sur les caractéristiques non-divulguées des utilisateurs ([Porter et al., 2009]).

D'autre part, l'exploration de bases de données importantes se faisant généralement par ordinateur, paralléliser le code en plusieurs processeurs ou CPUs est souvent nécessaire. Cependant, la transmission d'informations d'un processeur à un autre est un procédé généralement très lent. En conséquence, trouver une bonne subdivision du réseau est intéressant dans le sens où elle permet de minimiser les échanges entre CPUs ([Newman, 2010]).

La détection de communautés constitue donc une carte maîtresse dans l'analyse de grandes bases de données. C'est notamment pourquoi elle a connu un franc succès ces dernières années dans la science des réseaux.

1.4 Notions théoriques

Dans cette section, nous allons présenter plusieurs rappels sur les réseaux et leurs constituants ainsi que quelques notions importantes dans leur étude. Les notions élémentaires s'inspirent de [Lambiotte & Tabourier, 2013-2014], d'autres éléments théoriques proviennent de différents auteurs, cités alors dans les paragraphes concernés.

En mathématique, un réseau est constitué d'un ensemble de points appelés *sommets*. Si une certaine interaction existe entre eux, celle-ci est représentée par une *arête* reliant les sommets concernés. Dans la suite de ce mémoire, nous noterons n le nombre de sommets et m le nombre d'arêtes, ce qui correspond à la convention habituelle en littérature.

Une manière très simple de représenter le réseau se fait grâce à la matrice d'*adjacence*, notée A, qui est de dimension $n \times n$ et est définie telle que ses éléments

$$A_{ij} = \begin{cases} 1 \text{ s'il existe une arête entre les sommets } i \text{ et } j, \\ 0 \text{ sinon.} \end{cases}$$

De par sa définition, cette matrice capte toute la structure du réseau. Par exemple, sur le réseau de la FIGURE 1.1 composé de cinq sommets et cinq arêtes, la matrice d'adjacence est donc la matrice 5×5 définie par

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}.$$

Il est à noter que cette matrice est symétrique par définition et sa diagonale présente seulement des zéros car ce réseau ne contient pas de boucle (arête incidente à un seul sommet).

Chaque sommet peut être caractérisé par le nombre d'arêtes passant par celui-ci, c'est-à-dire son *degré*, que nous noterons, pour le sommet i, k_i dans la suite de ce mémoire. La matrice d'adjacence nous permet de trouver ce nombre par la relation suivante

$$k_i = \sum_{j=1}^{n} A_{ij}. \tag{1.1}$$

De plus, chaque arête ayant deux extrémités, la somme des degrés du réseau est égale au double du nombre total d'arêtes, c'est-à-dire

$$\sum_{i=1}^{n} k_i = 2m. \tag{1.2}$$

Nous utiliserons ce résultat à plusieurs reprises par la suite.

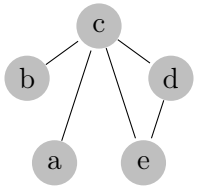

FIGURE 1.1: Exemple de réseau

Précédemment, nous avons présenté un élément important dans l'étude des réseaux, la matrice d'adjacence, permettant de capter la structure entière de ceux-ci. Il en existe une autre qui lui est liée, mais donnant plus d'information sur la structure du graphe, c'est la matrice *Laplacienne*. Cette dernière est définie par la relation $L = D - A$, où D est une matrice diagonale dont les éléments diagonaux sont les degrés des sommets. Cette matrice a des propriétés spectrales importantes. En effet, toutes ses valeurs propres sont positives. De plus, le vecteur $\mathbb{1} = (1, \ldots, 1)$ est toujours un vecteur propre de L associé à zéro, sa plus petite valeur propre. Enfin, le déterminant de cette matrice étant toujours nul dû à la valeur propre nulle, elle est singulière.

En ce qui a trait à la dynamique sur un réseau, elle peut s'établir via les marches aléatoires et les chaînes de Markov. Les éléments théoriques concernant ces deux notions proviennent des deux sources [Delvenne et al., 2009] et [Lambiotte & Masuda, 2016].

Il est possible de construire un processus dynamique grâce à l'utilisation de *marches aléatoires*. Cela consiste en des chemins construits sur le réseau. Initialement, un marcheur est posé sur un sommet quelconque de ce dernier. Ensuite, à chaque étape de sa marche, il choisit de manière uniformément aléatoire une des arêtes adjacentes à sa position et se déplace le long de celle-ci vers le sommet suivant. Généralement, le choix de la direction du déplacement étant aléatoire, le marcheur peut emprunter arêtes et sommets plus d'une fois. Les marcheurs aléatoires sont utilisés dans de nombreux domaines tels que l'observation des dynamiques de diffusion sur un réseau et la détection de communautés pour n'en citer que deux. Dans le cadre de ce mémoire, cette notion est couplée avec la suivante, les chaînes de Markov, au sein d'un type de méthode analysant la stabilité des communautés et que nous décrirons plus loin.

Les chaînes de Markov représentent une dynamique stochastique vérifiant les hypothèses de Markov pour un nombre discret d'états. Dans notre cas, un état serait l'ensemble des sommets identifiant la position des marcheurs aléatoires et ainsi le processus correspondrait à leur marche aléatoire sur le réseau. Usuellement, un état dans un processus stochastique dépend de l'ensemble de ceux qui l'ont précédé. Cependant, sous les hypothèses de Markov, la probabilité conditionnelle d'observer un état futur ne dépend que de l'état actuel. Mathématiquement dit, cela signifie que

$$P(X_{i+1} = x_{i+1} | X_i = x_i, \ldots, X_1 = x_1) \quad = \quad P(X_{i+1} = x_{i+1} | X_i = x_i),$$

où X_* est un état de la dynamique stochastique. Comme nous sommes sur un réseau et plus particulièrement dans une marche aléatoire, la probabilité conditionnelle définissant la chaîne de Markov équivaut à celle de quitter un sommet pour un autre adjacent à celui-ci.

Notre mémoire abordant également des méthodes spectrales, cette section ne serait pas complète sans un rappel théorique sur la structure propre des matrices. Considérons l'une de ces dernière, notée A, de dimension $N \times N$ et les notions théoriques présentées dans

[Lemaître, 2013-2014]. Un scalaire λ et un vecteur x vérifiant la relation suivante,

$$Ax = \lambda x,$$

sont respectivement appelés *valeur propre* et *vecteur propre* associé à celle-ci de la matrice A. Cette dernière possède au plus N valeurs propres et vecteurs propres associés. Si elle est symétrique en sus, l'entièreté de celles-ci seront réelles.

Un dernier point à éclaircir est le théorème de Perron-Frobenius utilisé dans le chapitre suivant. Ce théorème présente de nombreuses assertions vraies pour une matrice non négative et primitive mais, nous n'en énonçons qu'une. Celle-ci affirme l'existence et l'unicité d'un vecteur propre d'éléments strictement positifs. Le lecteur intéressé trouvera plus d'informations en [Meyer, 2000].

Ces notions indispensables maintenant définies, nous pouvons poursuivre ce chapitre par la synthèse des différents types de méthodes que nous analyserons pour la résolution du problème de détection de communautés.

1.5 Brève description des méthodes

Le but des méthodes décrites par la suite reste toujours le même, trouver les lignes de séparation naturelles du réseau. Autrement dit, identifier les groupes ou les communautés de sommets qui composent ce dernier de manière à minimiser le nombre d'arêtes entre ces groupes et ce, afin de comprendre la structure interne du réseau. Cependant, comme développé dans la section 1.1, le problème étant posé trop vaguement, il existe une grande variété de définitions différentes de celui-ci menant par conséquent à une grande variété d'algorithmes différents.

Dans cette section, nous nous attelons à décrire brièvement les différentes classes de méthodes analysées dans ce mémoire. Elles abordent le problème de détection de communauté de trois façons différentes, à savoir l'optimisation de la modularité, l'inférence statistique et l'optimisation de la stabilité. Ces méthodes seront ensuite détaillées dans le chapitre 2.

1.5.1 Optimisation de modularité

Ce type de méthodes, dont les éléments présentés dans cette sous-section s'inspirent de [Newman, 2010], est probablement le plus utilisé pour la détection de communautés. Il se base sur la notion de modularité pour caractériser la qualité d'une division de sommets du réseau. Cette notion se définit par la part du nombre total d'arêtes de la différence entre deux termes, le premier étant le nombre d'arêtes entre les sommets d'une même communauté, le deuxième correspondant à celui que nous espérons trouver si elles étaient placées aléatoirement sur le réseau. Pour le dire autrement, la modularité est une mesure de l'importance des connexions entre sommets au sein d'un même groupe. Elle est positive

s'il existe plus de liens qu'espéré par les probabilités dans un groupe et elle est toujours inférieure à 1. Elle est donnée par la formule suivante

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(g_i, g_j),$$

où m est le nombre total d'arêtes présentes sur le réseau et g_i et g_j représentent respectivement les groupes auxquels appartiennent les sommets i et j. De plus amples détails seront donnés quant à l'origine de cette formule dans la section 2.1 traitant de cette méthode.

Notons que cette méthode se base sur le nombre d'arêtes, non pas entre les groupes (correspondant à la taille de coupe) mais à l'intérieur de ceux-ci. Cependant, ces deux approches sont équivalentes car chaque arête liant deux communautés ne peut appartenir à une de ces dernières, la somme de la taille de coupe et du nombre d'arêtes internes aux communautés nous donne donc le nombre total d'arêtes composant le réseau.

De plus, la modularité est plus intéressante que la taille de coupe pour identifier une bonne division. En effet, si nous trouvons une subdivision du réseau telle qu'il existe peu d'arêtes entre les groupes de celle-ci mais que la taille de coupe est du même ordre que celle obtenue si nous avions placé les arêtes aléatoirement, alors nous n'aurions rien obtenu d'intéressant sur la structure du réseau et les communautés obtenues ne nous apporteraient rien. La modularité quant à elle tient compte de ce détail en calculant la qualité d'une subdivision relativement au modèle *neutre* de placement aléatoire d'arêtes. Donc, en maximisant la modularité, nous cherchons à trouver la quantité maximale d'arêtes dans un groupe de manière relative à celle espérée si celles-ci avaient été placées au hasard dans le réseau.

1.5.2 Inférence statistique

L'inférence statistique constitue également une méthode analysée dans ce mémoire. Elle consiste à faire concorder le réseau obtenu par un modèle génératif au réseau observé et les paramètres de cette concordance nous informeront sur la structure du réseau.

Il existe de nombreux modèles pour générer des réseaux. Nous trouvons, par exemple, le *modèle de blocs stochastiques* qui est le plus couramment utilisé dans le contexte de détection de communautés. Les notions théoriques associées à ce modèle générateur, de même que la brève explication de la méthode proviennent de [Newman, 2016]. Ce dernier produit un réseau divisé en q groupes dénotés par g_i pour un sommet i. Cela se fait via une matrice $q \times q$ notée Ω dont les éléments ω_{rs} représentent la probabilité de placer une arête entre deux sommets appartenant aux groupes r et s. Il est aisé de comprendre que les éléments diagonaux de cette matrice seront plus importants que les autres. En effet, dans un réseau modulaire, la probabilité de placer une arête entre deux sommets d'une même communauté doit être plus élevée qu'entre deux communautés différentes.

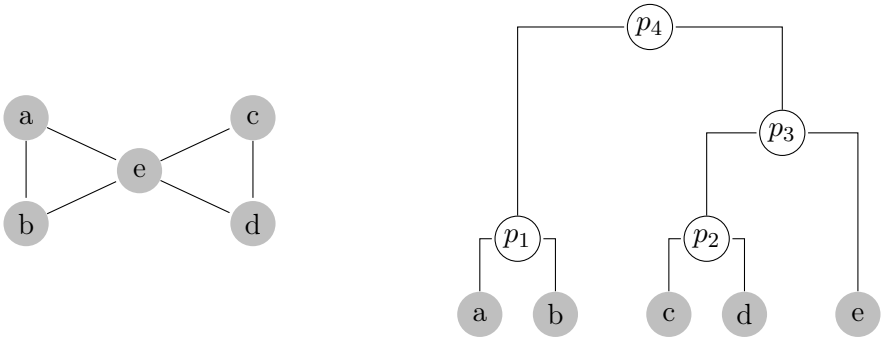

FIGURE 1.2: À gauche : exemple de réseau. À droite : exemple de dendrogramme correspondant.

Nous trouvons également un autre exemple de modèle pour générer les réseaux, le *regroupement hiérarchique*. Celui-ci est intéressant car il va au-delà du regroupement simple en communautés. Assurément, il inclut l'organisation du réseau à toutes échelles topologiques. Autrement dit, il prend en compte le fait qu'un groupe de sommets peut avoir plusieurs sous-groupes qui peuvent être composés à leur tour de plusieurs sous-sous-groupes... Les éléments théoriques présentés dans la suite de ce paragraphe et liés au regroupement hiérarchique se basent sur [Clauset et al., 2008] et [Clauset et al., 2008 (bis)], cette dernière source étant un complément d'information de la première. La structure hiérarchique d'un réseau est habituellement représentée par un arbre ou encore un *dendrogramme*. Dans ce dernier, chaque sommet du réseau représente une feuille de l'arbre et les paires de sommets liés par un chemin quelconque ont un ancêtre commun. Enfin, la hauteur de ces ancêtres dans l'arbre va de paire avec la longueur du chemin auquel ils sont associés. Afin de visualiser ce qu'est un dendrogramme, un exemple est donné à la FIGURE 1.2. Une telle représentation peut être utilisée pour modéliser un réseau en associant à chaque nœud interne r une probabilité p_r. Ainsi, une arête est placée entre deux sommets suivant la probabilité de leur plus proche ancêtre commun. Ce modèle est communément appelé un *graphe aléatoire hiérarchisé*. Étant donné un dendrogramme et un ensemble de probabilités associées aux nœuds internes de celui-ci, ce modèle nous permet de générer des réseaux avec une structure hiérarchisée. Pour capturer la structure modulaire, ce modèle pose des valeurs de probabilités élevées au niveau inférieur du dendrogramme et fait décroître celles-ci en remontant vers le sommet.

Bien que ces modèles aient pour but premier la génération de réseaux aléatoires, lorsqu'ils sont appliqués au problème de détection de communautés, ils sont utilisés de manière différente. Dans le sens où, en les faisant correspondre au réseau donné, ils servent à déduire la structure de celui-ci. Cette classe de méthodes consiste donc à trouver les paramètres maximisant le logarithme de la probabilité, appelé le *log-likelihood*, que le réseau observé ait été généré par le modèle utilisé. De cette façon, le découpage en communautés du réseau généré maximisant ce log-likelihood nous donne la structure modulaire du réseau observé.

1.5.3 Stabilité des communautés

Contrairement aux autres classes de méthodes, celle-ci fait intervenir une mesure de la qualité des divisions du réseau valable sur plusieurs échelles de temps, la stabilité. Pour ce faire, elle utilise « la relation établie entre les graphes et les chaines de Markov. Cette dernière stipule qu'à chaque graphe non pondéré nous pouvons associer une marche aléatoire dans laquelle la probabilité de quitter un sommet est uniformément distribuée parmi les arêtes sortantes » ([Delvenne et al., 2009]).

Sur base des notions de marches aléatoires et de chaînes de Markov précédemment vues dans la section 1.4 et des sources [Delvenne et al., 2009] et [Lambiotte & Masuda, 2016], nous pouvons définir *la stabilité* qui nous servira de mesure de la qualité d'un partitionnement du réseau dans cette méthode. La stabilité décrit si le marcheur aléatoire a tendance à rester dans la même communauté au fil du temps. Autrement dit, une partition des sommets du réseau sera bonne sur une certaine échelle de temps si l'état initial de la chaîne de Markov a une grande probabilité de rester dans la communauté de départ.

Cela apporte une interprétation dynamique des communautés. En effet, à une échelle de temps donnée, elles correspondent à des bassins de stabilité dynamique. Autrement dit, ce sont des zones dont la probabilité d'en sortir est très faible due au nombre important de connexions internes.

1.6 Conclusion

Nous avons vu à la section 1.2 l'apport de nombreux arguments pour comprendre l'existence des communautés dans un réseau. Elles permettent l'adaptation rapide aux changements extérieurs, le traitement des informations localisé et plus efficace, une diffusion plus rapide sur le réseau, la division des sommets de ce dernier tout en le laissant connexe, l'évolution sous un milieu changeant et enfin, elles seraient associées au développement du réseau en lui-même. Conséquemment, les étudier n'est pas anodin.

De plus, formant la structure mésoscopique du graphe, nous avons précisé à la section 1.3 l'utilité des communautés dans analyse de l'organisation interne de ce dernier. Quant à l'exploration de données via un ordinateur, lorsque la quantité d'information est trop importante, celles-ci peuvent également servir la parallélisation de code en plusieurs processeurs.

Enfin, nous avons vu que, de par leur nature interdisciplinaire, du problème NP-complet de leur détection et de la spécificité de leur définition au domaine d'étude, les communautés et leur détection ont grandement suscité l'attention de chercheurs de tous horizons. Un nombre important de méthodes se sont ainsi développées, il nous est donc impossible de toutes les considérer dans ce mémoire. À la dernière section de ce chapitre, nous avons dès lors introduit les classes de méthodes détaillées et analysées dans le chapitre suivant.

Chapitre 2

Développement des méthodes vers une version spectrale

Ce chapitre porte sur l'analyse théorique des différentes méthodes précédemment introduites. Celles-ci abordent chacune une manière spécifique de traiter le problème de détection de communautés, à savoir l'optimisation de la modularité, l'inférence statistique et l'optimisation de la stabilité. Elles seront en premier lieu formellement définies selon leurs sources respectivement citées en début de section. Ensuite, notre intuition plaçant le point commun entre toutes ces méthodes sur les outils spectraux, nous développerons un lien entre celles-ci et une méthode de détection, dite *spectrale*, utilisant ces outils.

2.1 Optimisation de la modularité

Ce type de méthodes vise à maximiser une certaine mesure de la qualité des communautés d'un réseau, la modularité. Celle-ci sera définie formellement avant de présenter une méthode spectrale pour l'optimiser. La description et le développement de l'ensemble des éléments de cette section se basent sur [Newman, 2010].

2.1.1 Définition formelle de la modularité

La modularité fut précédemment introduite à la section 1.5.1. Pour rappel, elle se base sur le nombre d'arêtes au sein d'un groupe et le compare avec le nombre espéré si elles étaient placées aléatoirement.

Pour une définition plus formelle, supposons que notre réseau de n sommets et m arêtes soit divisé en q groupes et notons g_i le groupe auquel appartient le sommet i. Avec ces notations, nous avons que le nombre total d'arêtes existant entre les sommets d'un même groupe est défini par

$$A\_ex = \sum_{\substack{\text{arêtes} \\ \text{liant } i \text{ à } j}} \delta(g_i, g_j),$$

où $\delta(g_i, g_j)$ correspond au symbole de Kronecker valant 1 si g_i est égal à g_j (si les sommets i et j appartiennent au même groupe) et étant nul sinon. L'information concernant les arêtes liant i à j est reprise dans la matrice d'adjacence du réseau. Dès lors, le nombre total d'arêtes existant entre les sommets d'un groupe s'exprime comme suit,

$$A\_ex = \frac{1}{2} \sum_{ij} A_{ij} \delta(g_i, g_j). \tag{2.1}$$

Dans cette équation, le facteur $\frac{1}{2}$ contrecarre le fait que les arêtes sont comptées deux fois dans la matrice d'adjacence.

À présent, calculons le nombre d'arêtes que nous espérons trouver si celles-ci étaient placées aléatoirement sur le réseau. Supposons qu'une des extrémités d'une arête soit positionnée sur le sommet i de degré k_i. La probabilité que son autre extrémité soit attachée au sommet j de degré k_j est donc de $k_j/2m$. Effectivement, le nombre d'arêtes sur l'entièreté du réseau étant égal à m, ce dernier contient au total $2m$ extrémités. En suivant ce raisonnement pour toutes les k_i arêtes liées au nœud i, le nombre d'arêtes espéré entre les sommets i et j est de $k_i k_j/2m$. Par voie de conséquence, l'utilisation du symbole de Kronecker nous permet d'atteindre la quantité d'arêtes placées aléatoirement entre les sommets d'un même groupe. Ceci s'écrit par l'équation suivante,

$$A\_esp \;=\; \frac{1}{2}\sum_{ij}\frac{k_i k_j}{2m}\delta(g_i, g_j). \tag{2.2}$$

Dès lors, nous pouvons calculer la différence entre les deux quantités A_ex (2.1) et A_esp (2.2) et la diviser ensuite par le nombre total d'arêtes pour obtenir notre définition de la modularité, notée Q, nous avons donc que

$$Q = \frac{1}{m}\left(\frac{1}{2}\sum_{ij}A_{ij}\delta(g_i, g_j) - \frac{1}{2}\sum_{ij}\frac{k_i k_j}{2m}\delta(g_i, g_j)\right).$$

Par la mise en évidence du facteur $\frac{1}{2}$ et du symbole de Kronecker, nous obtenons finalement l'expression suivante,

$$Q = \frac{1}{2m}\sum_{ij}\left(A_{ij} - \frac{k_i k_j}{2m}\right)\delta(g_i, g_j). \tag{2.3}$$

Cette définition de la modularité ne semble pas convenir à tous les réseaux comme l'ont montré [Fortunato & Barthélemy, 2006]. En effet, son optimisation ne détecte pas les communautés contenant un nombre d'arêtes de l'ordre de $\sqrt{2m}$ ou plus petit. Cela vient du deuxième terme de la somme dans la définition ci-avant (2.3). Si le nombre d'arêtes du réseau augmente, alors $\frac{k_i k_j}{2m}$ devient petit et le premier terme interne à la somme prend plus d'importance. Conséquemment, la modularité est maximisée si de nombreux sommets se trouvent dans la même communauté. Ainsi, ce maximum va rassembler des petits groupes de sommets en un seul et ainsi échouer dans la détection de ces derniers. La division du réseau obtenue ne correspondra donc pas à la division réelle. Pour pallier à ce problème, la fonction de modularité généralisée a été définie comme suit,

$$Q(\gamma) = \frac{1}{2m}\sum_{ij}\left(A_{ij} - \gamma\frac{k_i k_j}{2m}\right)\delta(g_i, g_j), \tag{2.4}$$

où le paramètre γ sert de paramètre de résolution dans le sens où il va équilibrer les deux termes A_{ij} et $\frac{k_i k_j}{2m}$. Autrement dit, lorsque $\gamma = 1$, la modularité traditionnelle (2.3) est retrouvée. S'il est plus petit ou plus grand, cela impliquera, respectivement, la découverte de communautés plus larges ou plus petites.

2.1.2 Méthode spectrale

Une fois la notion de modularité bien définie, il nous reste à trouver la valeur qui l'optimise en fonction des variables g_* définissant les communautés. Malheureusement, « ce problème est NP-complet » ([Brandes et al.,2008]) mais il a tout de même inspiré un grand nombre de méthodes pour parvenir à sa résolution. Nous allons garder notre attention sur la méthode spectrale en établissant un lien entre une méthode d'optimisation de la modularité et les outils spectraux liés à celle-ci. Le travail effectué dans cette section se base sur [Newman, 2010].

Bissection du réseau

En premier lieu, nous considérons le problème le plus facile, à savoir la séparation du réseau en deux groupes a et b. Le cas général sera vu plus loin dans cette section. De plus, dans un souci de simplification, nous utilisons la définition de modularité simple (2.3).

Soit notre réseau constitué de n sommets et m arêtes, nous définissons des quantités s_* de manière à représenter la bissection de celui-ci. Aussi s_i vaudra-t-elle $+1$ si le sommet i appartient au groupe a et -1 sinon. Remarquons que

$$\frac{1}{2}(s_i s_j + 1) = \begin{cases} 1 \text{ si } i \text{ et } j \text{ appartiennent au même groupe,} \\ 0 \text{ sinon.} \end{cases}$$

Partant de ce fait, nous pouvons utiliser le symbole de Kronecker pour transformer cette relation de la façon suivante,

$$\delta(g_i, g_j) = \frac{1}{2}(s_i s_j + 1). \tag{2.5}$$

Grâce à cette égalité, nous pouvons reformuler la définition de modularité (2.3) comme suit,

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \frac{1}{2}(s_i s_j + 1),$$

ou encore, en mettant en évidence le facteur $\frac{1}{2}$,

$$Q = \frac{1}{4m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) (s_i s_j + 1). \tag{2.6}$$

Nous introduisons à présent la matrice B dont chaque élément est défini par la relation suivante,

$$B_{ij} = A_{ij} - \frac{k_i k_j}{2m}.$$

Cette matrice, bien connue dans la littérature, est appelée *matrice de modularité* et en l'utilisant pour transformer l'équation 2.6, nous obtenons que

$$Q = \frac{1}{4m} \sum_{ij} B_{ij}(s_i s_j + 1).$$

En distribuant les termes internes à la somme, nous pouvons écrire la modularité de la manière suivante,

$$Q = \frac{1}{4m}\left(\sum_{i,j} B_{ij}s_i s_j + \sum_{i,j} B_{ij}\right).$$

Cette distribution n'est pas vide de sens car nous allons pouvoir simplifier encore cette expression. En effet, compte tenu de la propriété liant la matrice d'adjacence aux degrés des sommets (1.1, page 7), la matrice de modularité vérifie, pour la somme sur i comme sur j, que

$$\sum_i B_{ij} = \sum_i A_{ij} - \frac{k_j}{2m}\sum_i k_i = k_j - \frac{k_j}{2m}2m = 0.$$

Ainsi, nous obtenons finalement une nouvelle expression pour la modularité lorsque nous considérons une bissection du réseau, à savoir

$$Q = \frac{1}{4m}\sum_{i,j} B_{ij}s_i s_j$$

ou encore, en notation matricielle,

$$Q = \frac{1}{4m}s^T B s, \tag{2.7}$$

où s est le vecteur composé des éléments s_i.

Conséquemment, notre but est de trouver le vecteur s maximisant l'équation (2.7) avec, comme seule contrainte, ses éléments s_i uniquement restreints à ± 1. Cette contrainte étant assez forte, nous procédons donc à une relaxation de celle-ci en imposant au vecteur s de vérifier

$$s^T s = \sum_i s_i^2 = n. \tag{2.8}$$

Autrement dit, nous permettons au vecteur s de contenir n'importe quels éléments pourvu que sa norme soit identique à celle d'un vecteur composé uniquement de ± 1.

Pour résoudre ce problème d'optimisation, nous allons utiliser un multiplicateur de Lagrange, noté β, associé à la contrainte (2.8) citée ci-avant. L'équation d'optimisation à résoudre est équivalente à

$$\frac{\partial}{\partial s_i}\left(\sum_{k,j} B_{kj}s_k s_j + \beta\left(n - \sum_k s_k^2\right)\right) = 0$$

et ce, pour $i = 1 \ldots n$. En effectuant la dérivée partielle du membre de gauche, nous arrivons à l'égalité suivante,

$$2\sum_j B_{ij}s_j - 2\beta s_i \ = 0.$$

Les vecteurs Bs et βs doivent ainsi être égaux et de ce fait, le vecteur s doit être un vecteur propre de la matrice B associé à la valeur propre β. En utilisant cette relation dans l'équation (2.7), nous pouvons écrire que

$$Q = \frac{1}{4m}\beta s^T s. \tag{2.9}$$

Par la contrainte imposée (2.8), nous aboutissons finalement la valeur suivante pour la modularité,

$$Q = \frac{n}{4m}\beta. \tag{2.10}$$

Pour maximiser la modularité Q, il nous suffit donc de subdiviser le réseau selon les éléments du vecteur s. Par la relation finale obtenue lors du développement ci-avant (2.10), ce dernier est choisi comme étant le vecteur propre correspondant à la plus grande valeur propre β de la matrice de modularité B. Cependant, comme s ne peut prendre que les valeurs ± 1, nous ne pouvons pas le définir comme tel mais nous allons le prendre le plus proche possible de ce vecteur propre en maximisant le produit scalaire suivant

$$s^T x = \sum_i s_i x_i,$$

où x est le vecteur propre associé à la plus grande valeur propre de B. Ce produit est maximisé lorsque les éléments de s associés aux valeurs positives et négatives de x valent respectivement $+1$ et -1.

Dans le cas de la maximisation de la modularité généralisée, la seule modification est apportée à la matrice B dont nous devons trouver le vecteur propre associé à sa plus grande valeur propre. Cette matrice a alors ses composantes définies comme suit

$$B_{ij} = A_{ij} - \gamma \frac{k_i k_j}{2m},$$

où γ est une constante.

Recherche des communautés naturelles du réseau

Nous venons de voir une méthode pour diviser le graphe en seulement deux groupes, mais, notre but étant la découverte des communautés naturelles du réseau, il est fort probable que celles-ci soient plus nombreuses. Un moyen d'en obtenir davantage serait d'effectuer des bissections successives du graphe. Plus explicitement, en partant de deux communautés, diviser l'une d'elles en deux permet d'en avoir trois, diviser l'une de ces trois-ci en deux permet d'en avoir quatre et ainsi de suite. Cependant, le nombre de communautés n'est pas fixé à l'avance par le chercheur. Aussi procéderons-nous à des bissections successives tout en évaluant le changement dans la modularité apporté par une division supplémentaire. De cette façon, nous pourrons analyser l'évolution de la modularité en fonction du nombre de groupes présents dans notre réseau.

Considérons le changement ΔQ de modularité du réseau entier après une bissection supplémentaire de la communauté c de taille n_c. Il est donné par

$$\Delta Q = \frac{1}{2m} \sum_{i,j \in c} B_{ij} \delta(g_i, g_j) - \frac{1}{2m} \sum_{i,j \in c} B_{ij},$$

où le premier terme représente la modularité relative aux arêtes du groupe c subdivisé. Le deuxième terme, quant à lui, reprend cette même modularité avant toute bissection de ce groupe. Dès lors, le symbole de Kronecker associé à ce terme par la définition de cette mesure de qualité (2.3, page 14) sera systématiquement égal à l'unité. Nous l'avons donc négligé dans les notations afin de simplifier l'écriture. En utilisant de nouveau la relation 2.5 (page 15), nous pouvons réécrire le changement ΔQ comme suit,

$$\Delta Q = \frac{1}{2m} \left(\frac{1}{2} \sum_{i,j \ \in \ c} B_{ij}(s_i s_j + 1) - \sum_{i,j \ \in \ c} B_{ij} \right).$$

En développant le premier terme puis en simplifiant, nous arrivons à l'expression suivante,

$$\Delta Q = \frac{1}{4m} \left(\sum_{i,j \ \in \ c} B_{ij} s_i s_j - \sum_{i,j \ \in \ c} B_{ij} \right),$$

Comme les éléments s_i valent ± 1 par définition, le carré de s_i vaut 1 pour toute valeur de i. Dès lors, nous pouvons écrire que

$$Q = \frac{1}{4m} \left(\sum_{i,j \ \in \ c} B_{ij} s_i s_j - \sum_{i,j \ \in \ c} \delta_{ij} s_i s_j \sum_{k \ \in \ c} B_{ik} \right).$$

En rassemblant les sommes et en mettant en évidence le produit $s_i s_j$, nous obtenons que

$$\Delta Q = \frac{1}{4m} \sum_{i,j \ \in \ c} \left(B_{ij} - \delta_{ij} \sum_{k \ \in \ c} B_{ik} \right) s_i s_j,$$

ou encore, en notation matricielle,

$$\Delta Q = \frac{1}{4m} s^T B^{(c)} s,$$

où $B^{(c)}$ est la matrice de taille $n_c \times n_c$ dont ses éléments sont les $B_{ij} - \delta_{ij} \sum_k B_{ik}$. Nous arrivons finalement à une expression semblable à l'équation 2.7 (page 16). Ainsi, pour résoudre le problème de son optimisation, il suffit de reproduire le développement réalisé précédemment. Le contrat est bien rempli, cette méthode de recherche du maximum de modularité fait bien intervenir les outils spectraux liés à une des matrices définissant le réseau.

2.2 Inférence statistique

Ce type de méthodes consiste à faire concorder le réseau obtenu par un modèle génératif au réseau observé. De cette façon, les paramètres de cette concordance nous renseigneront

sur la structure du réseau et ainsi, s'il est modulaire, nous donner des informations sur les communautés de celui-ci. Nous définirons en premier lieu le modèle génératif considéré dans ce mémoire pour ensuite nous atteler à trouver un lien entre ce modèle et les outils spectraux. L'ensemble du travail sur cette section se base sur [Newman, 2016].

2.2.1 Méthode d'inférence statistique sur le modèle de blocs stochastiques

Comme l'annonce le titre de cette sous-section, le modèle que nous allons détailler est celui dit « en blocs stochastiques ». Une définition formelle de ce dernier est offerte dans laquelle le modèle le plus basique de cette théorie sera explicité puis amélioré jusqu'à obtenir un cas très général convenant à la génération de tout réseau. Ensuite, nous définirons l'inférence statistique basée sur ce dernier modèle.

Modèle de blocs stochastiques

Comme précédemment, prenons un nombre n de sommets et classons ces derniers en q groupes. À nouveau, nous notons g_i le groupe auquel appartient le sommet i. Dans le modèle de blocs stochastiques classique, les arêtes sont placées aléatoirement avec chacune une certaine probabilité ω_{rs}. Cette dernière dépend des groupes r et s auxquels appartiennent respectivement les deux sommets liés par cette arête. Ceci correspond à une approche basée sur une distribution de Bernoulli.

Cependant, la méthode de détection de communautés travaille généralement avec une autre formulation de ce modèle. En effet, ce n'est pas une seule arête qui est placée mais un certain nombre selon une distribution de Poisson de moyenne ω_{rs}. Cette formulation peut permettre au réseau de contenir plusieurs arêtes entre deux mêmes sommets, ce que nous appelons une *arête multiple*. Il est possible en sus de permettre au réseau de contenir des *boucles*, c'est-à-dire des arêtes incidentes à un seul sommet. Dans ce dernier cas, le nombre de boucles suit une distribution de Poisson de moyenne $\frac{1}{2}\omega_{rr}$, où le facteur $\frac{1}{2}$ est ajouté pour des raisons de simplifications algébriques. Il est à noter que la plupart des réseaux rencontrés dans le monde réel ne contiennent ni arêtes multiples, ni boucles. Toutefois, ces réseaux sont également fort clairsemés impliquant une faible densité d'arêtes et de petites valeurs pour les paramètres ω_{rs}. Par conséquent, la probabilité de placer des boucles et des arêtes multiples sera elle-même très faible. Aussi est-il équivalent d'utiliser une distribution de Poisson ou de Bernoulli, la dernière étant présentée au début de cette sous-section. Cependant, « la première reste techniquement plus facile à utiliser » ([Newman, 2016]).

Conservons l'approche de Poisson, nous pouvons construire une matrice de dimension $q \times q$ dont ses éléments sont les paramètres ω_{rs}. Remarquons que si les éléments diagonaux ω_{rr} sont plus grands que les éléments non-diagonaux, alors le réseau généré par ce modèle aura plus de chance de présenter une structure modulaire. En effet, le nombre d'arêtes à l'intérieur des groupes aura tendance à être plus élevé qu'entre ceux-ci.

Les paramètres utilisés pour faire correspondre le réseau généré au réseau réel seront donc les membres des groupes g_* ainsi que les valeurs ω_{**} contenues dans la matrice établie par ce modèle. Cependant, en l'état, ce dernier ne tient pas compte des degrés des sommets du réseau réel. En effet, ceux du réseau généré ont leur degré suivant une distribution de Poisson. Par conséquent, nous devons utiliser un modèle légèrement différent pouvant coïncider avec n'importe quelle distribution de degrés du réseau. Nous savons que le nombre d'arêtes espéré entre les sommets i et j dans le réseau observé vaut $k_i k_j/2m$ [1]. Nous allons quantifier cette valeur avec les paramètres ω_{rs} de manière à obtenir une probabilité tenant compte des degrés du réseau observé. Ainsi, les arêtes seront placées de manière aléatoire suivant une distribution de Poisson de moyenne $(k_i k_j/2m)\omega_{g_i g_j}$ et de la moitié de cette valeur pour les boucles.

Inférence statistique

Comme précisé ci-avant, les paramètres servant à faire correspondre le réseau généré par le modèle au réseau observé sont g_* et ω_{**}. En leur donnant une certaine valeur, nous pouvons exprimer la probabilité que le réseau observé ait été effectivement engendré par le modèle. Conséquemment, par la maximisation de celle-ci, nous serons à même de déterminer les paramètres correspondant le mieux à la structure interne du réseau observé.

Cette probabilité est notée $P(A|\Omega, g)$, où Ω est la matrice contenant les valeurs des paramètres ω_{rs} et g est le vecteur dont les composantes sont les éléments g_i. En faisant intervenir les paramètres ω_{**}, nous obtenons la définition suivante pour cette probabilité,

$$P(A|\Omega, g) = \prod_i P(A_{ii}|\omega_{g_i g_i}) \prod_{i<j} P(A_{ij}|\omega_{g_i g_j}),$$

où la distinction entre les boucles et les arêtes a été faite, ceci prenant tout son sens dans l'équation suivante. Notons également l'absence d'un produit sur les valeurs de i plus grandes que celles de j. Le réseau étant conventionnellement considéré non-dirigé [2], ce produit donnerait des informations redondantes. En se rappelant que les paramètres ω_{**} représentent la moyenne de la distribution de Poisson définie précédemment, nous pouvons écrire que

$$P(A|\Omega, g) = \prod_i \frac{\left(\frac{k_i^2}{4m}\omega_{g_i g_i}\right)^{A_{ii}/2}}{\left(\frac{1}{2}A_{ii}\right)!} e^{-(k_i^2/4m)\omega_{g_i g_i}} \prod_{i<j} \frac{\left(\frac{k_i k_j}{2m}\omega_{g_i g_j}\right)^{A_{ij}}}{A_{ij}!} e^{-(k_i k_j/2m)\omega_{g_i g_j}}.$$

Le facteur $\frac{1}{2}$ associé aux éléments A_{ii} dans l'exposant et au dénominateur de la distribution de Poisson sont dû à la convention adoptée donnant la valeur de 2 pour les boucles dans la matrice d'adjacence.

1. Voir, en page 14, le raisonnement menant à l'équation 2.2.
2. Réseau dans lequel les arêtes n'ont pas de direction.

Le logarithme de cette probabilité, aussi appelé *log-likelihood*, est donc défini par

$$\begin{aligned}\log P(A|\Omega, g) &= \sum_i \left(\frac{1}{2}A_{ii}\log\left(\frac{k_i^2}{4m}\omega_{g_ig_i}\right) - \frac{k_i^2}{4m}\omega_{g_ig_i} - \log\left(\frac{1}{2}A_{ii}!\right)\right) \\ &\quad + \sum_{i<j}\left(A_{ij}\log\left(\frac{k_ik_j}{2m}\omega_{g_ig_j}\right) - \frac{k_ik_j}{2m}\omega_{g_ig_j} - \log\left(A_{ij}!\right)\right).\end{aligned}$$

En développant le premier terme de chaque somme, nous pouvons dévoiler, respectivement, les éléments $\frac{1}{2}A_{ii}\log(\frac{k_i^2}{4m})$ et $A_{ij}\log(\frac{k_ik_j}{2m})$. Ces derniers étant constants par rapport aux paramètres d'optimisation, ils seront donc négligés ainsi que les deux termes $\log(\frac{1}{2}A_{ii}!)$ et $\log(A_{ij}!)$. Après ces simplifications, nous obtenons que

$$\log P(A|\Omega, g) = \frac{1}{2}\sum_i\left(A_{ii}\log(\omega_{g_ig_i}) - \frac{k_i^2}{2m}\omega_{g_ig_i}\right) + \sum_{i<j}\left(A_{ij}\log(\omega_{g_ig_j}) - \frac{k_ik_j}{2m}\omega_{g_ig_j}\right).$$

Par une mise en évidence du facteur $\frac{1}{2}$ pour les deux termes, la somme sur les valeurs de i plus petites que celles de j se voit doublée. Cette redondance nous permet alors de faire apparaître une somme sur i plus grand que j. De cette façon, nous pouvons rassembler ces trois termes en une unique somme. En fin de compte, le log-likelihood que le réseau généré correspond au réseau observé est défini par

$$\log P(A|\Omega, g) = \frac{1}{2}\sum_{ij}\left(A_{ij}\log(\omega_{g_ig_j}) - \frac{k_ik_j}{2m}\omega_{g_ig_j}\right). \tag{2.11}$$

Pour identifier les communautés du réseau, il suffit donc de maximiser cette quantité par rapport aux paramètres Ω et g. Par voie de conséquence, cela nous permettra de déterminer ceux qui correspondent le mieux au réseau observé. Il existe évidemment plusieurs manière de réaliser cette optimisation, mais nous allons nous concentrer sur une méthode faisant intervenir les outils spectraux.

2.2.2 Méthode spectrale

Pour mettre en lumière une méthode spectrale, nous allons faire apparaître une équivalence entre la modularité (2.3, page 14) et un cas particulier du modèle en blocs stochastiques décrit ci-avant, à savoir le *modèle de partition implantée* (ou encore *planted partition model*). Dans ce dernier, les paramètres ω_{rs} ont seulement deux valeurs possibles identifiées par ω_{in} si r et s correspondent à la même communauté et ω_{out} dans le cas contraire. « Ce modèle est moins flexible que celui décrit à la section 2.2.1, dans le sens où il impose aux groupes d'avoir la même proportion d'arêtes en leur sein et entre eux » ([Newman, 2016]).

Avant de développer l'équation (2.11) avec les données de ce cas particulier, remarquons que la nouvelle définition des paramètres ω_{rs} donne les deux relations suivantes :

$$\begin{aligned}\omega_{rs} &= (\omega_{\text{in}} - \omega_{\text{out}})\delta_{rs} + \omega_{\text{out}}, \\ \log\omega_{rs} &= (\log\omega_{\text{in}} - \log\omega_{\text{out}})\delta_{rs} + \log\omega_{\text{out}},\end{aligned}$$

où δ_{rs} est, comme à l'accoutumée, le symbole de Kronecker. La deuxième relation vient du fait que le $\log\omega_{rs}$ vaut le $\log\omega_{\text{in}}$ si r représente la même communauté que s et vaut le

$\log \omega_{\text{out}}$ sinon. À présent, substituons ces deux relations dans l'équation (2.11). Cela nous mène à la définition du log-likelihood pour le modèle de partition implantée avec correction des degrés, à savoir

$$\begin{aligned}\log P(A|\Omega, g) &= \frac{1}{2}\sum_{ij}\Bigg[A_{ij}\left((\log\omega_{\text{in}} - \log\omega_{\text{out}})\delta_{g_i g_j} + \log\omega_{\text{out}}\right) \\ &\quad -\frac{k_i k_j}{2m}\left((\omega_{\text{in}} - \omega_{\text{out}})\delta_{g_i g_j} + \omega_{\text{out}}\right)\Bigg].\end{aligned}$$

En distribuant les divers termes, en utilisant la propriété de la différence de logarithmes et en regroupant les termes dépendant des paramètres g_*, nous arrivons à ce que

$$\begin{aligned}\log P(A|\Omega, g) &= \frac{1}{2}\sum_{ij}\left[A_{ij}\log\left(\frac{\omega_{\text{in}}}{\omega_{\text{out}}}\right)\delta_{g_i g_j} - \frac{k_i k_j}{2m}(\omega_{\text{in}} - \omega_{\text{out}})\delta_{g_i g_j}\right] \\ &\quad +\frac{1}{2}\sum_{ij}\left[A_{ij}\log\omega_{\text{out}} - \frac{k_i k_j}{2m}\omega_{\text{out}}\right]. \qquad (2.12)\end{aligned}$$

Rappelons-nous que $\sum_{ij} A_{ij} = 2m$ tout comme $\sum_l k_l$. Nous pouvons donc simplifier la deuxième somme de l'équation (2.12) comme suit,

$$\frac{1}{2}\sum_{ij}\left[A_{ij}\log\omega_{\text{out}} - \frac{k_i k_j}{2m}\omega_{\text{out}}\right] = m(\log\omega_{\text{out}} - \omega_{\text{out}}).$$

La première somme quant à elle peut être retravaillée par la mise en évidence de $\log(\frac{\omega_{\text{in}}}{\omega_{\text{out}}})$ et de $\delta_{g_i g_j}$. Nous arrivons donc à l'égalité suivante,

$$\begin{aligned}&\frac{1}{2}\sum_{ij}\left[A_{ij}\log\left(\frac{\omega_{\text{in}}}{\omega_{\text{out}}}\right)\delta_{g_i g_j} - \frac{k_i k_j}{2m}(\omega_{\text{in}} - \omega_{\text{out}})\delta_{g_i g_j}\right] \\ &= \frac{1}{2}\log\left(\frac{\omega_{\text{in}}}{\omega_{\text{out}}}\right)\sum_{ij}\left[\left(A_{ij} - \frac{k_i k_j}{2m}\frac{(\omega_{\text{in}} - \omega_{\text{out}})}{(\log\omega_{\text{in}} - \log\omega_{\text{out}})}\right)\delta_{g_i g_j}\right].\end{aligned}$$

Le log-likelihood pour le modèle de partition implantée avec correction des degrés devient finalement

$$\begin{aligned}\log P(A|\Omega, g) &= \frac{1}{2}\log\left(\frac{\omega_{\text{in}}}{\omega_{\text{out}}}\right)\sum_{ij}\left[\left(A_{ij} - \frac{(\omega_{\text{in}} - \omega_{\text{out}})}{(\log\omega_{\text{in}} - \log\omega_{\text{out}})}\frac{k_i k_j}{2m}\right)\delta_{g_i g_j}\right] \\ &\quad +m(\log\omega_{\text{out}} - \omega_{\text{out}}).\end{aligned}$$

Pour simplifier l'écriture, utilisons les notations C, D et γ pour les constantes par rapport à g. Nous définissons ainsi

- $\star\ C = m\log\left(\frac{\omega_{\text{in}}}{\omega_{\text{out}}}\right)$
- $\star\ D = m(\log\omega_{\text{out}} - \omega_{\text{out}})$
- $\star\ \gamma = \frac{(\omega_{\text{in}} - \omega_{\text{out}})}{(\log\omega_{\text{in}} - \log\omega_{\text{out}})}$

De cette façon, le log-likelihood peut s'écrire tel que

$$\log P(A|\Omega, g) = C\frac{1}{2m}\sum_{ij}\left[\left(A_{ij} - \gamma\frac{k_i k_j}{2m}\right)\delta_{g_i g_j}\right] + D. \tag{2.13}$$

Cette expression est équivalente, à quelques constantes près disparaissant lors de la maximisation, à la modularité généralisée (2.4, page 14), qui, pour rappel, était définie comme suit

$$Q(\gamma) = \frac{1}{2m}\sum_{ij}\left(A_{ij} - \gamma\frac{k_i k_j}{2m}\right)\delta(g_i, g_j).$$

Par conséquent, si nous supposons que les paramètres ω_{in} et ω_{out} ont déjà été trouvés, maximiser le log-likelihood du modèle de partition implantée revient à maximiser la modularité généralisée.

Cependant, un problème subsiste. En effet, cette équivalence est atteinte pour les valeurs correctes de ω_{in} et ω_{out} déjà identifiées. Or, nous ne les connaissons pas avant la génération du réseau. Aussi la source [Newman, 2016] propose-t-elle d'utiliser une méthode itérative dans laquelle une première estimation de γ est effectuée permettant de générer le réseau et de trouver les communautés associées. À partir de ce réseau, les paramètres ω_{in} et ω_{out} peuvent être estimés donnant une nouvelle valeur à γ et ainsi de suite jusqu'à atteindre une convergence. Toutefois, l'article mentionne également que cette dernière n'est pas garantie pour des réseaux réels.

2.3 Optimisation de la stabilité

Comme précisé à la section 1.5.3, cette méthode se distingue des deux autres car elle fait intervenir une mesure de la qualité des divisions du réseau valable sur plusieurs échelles de temps. Dans un premier temps, nous définirons donc formellement cette mesure qu'est la stabilité des communautés. Nous chercherons ensuite le lien entre celle-ci et les méthodes spectrales de recherche de communautés. Les éléments théoriques présentés dans cette section s'inspirent des deux sources [Delvenne et al., 2009] et [Lambiotte & Masuda, 2016].

2.3.1 Définition formelle de la stabilité

Conventionnellement, nous considérons un réseau non-dirigé de n sommets et m arêtes. Définissons le vecteur $d = (k_1, \ldots, k_n)$ composé des degrés k_* des différents sommets du graphe. Par la relation (1.1, page 7), nous pouvons écrire que

$$d = A\mathbb{1},$$

où A est la matrice d'adjacence et $\mathbb{1}$ est le vecteur dont chaque composante vaut 1. Par la suite, nous utiliserons également la matrice diagonale $D = \text{diag}(d)$.

Comme énoncé dans les rappels à la section 1.4, une marche aléatoire sur un réseau définit une chaîne de Markov où la probabilité pour un marcheur aléatoire de quitter un

sommet se divise uniformément entre toutes les arêtes incidentes à celui-ci. Par conséquent, la probabilité de se déplacer d'un nœud i à un nœud j est définie par $T_{ij} = A_{ij}/k_i$. Nous pouvons dès lors définir une matrice $n \times n$, dite de *transition*, comme étant composée des éléments T_{ij}, c'est-à-dire $T = D^{-1}A$. Nous notons $p_{1,t},\ \ldots,\ p_{n,t}$ les probabilités d'atteindre chacun des n sommets du réseau au temps t. De cette façon et sous les hypothèses de Markov [3], ces probabilités se définissent comme suit,

$$p_{j,t+1} = \sum_{i=1}^{n} T_{ji} p_{i,t} \tag{2.14}$$

pour j allant de 1 jusqu'à n. En notation matricielle, nous pouvons écrire que

$$p_{t+1} = p_t T,$$

où $p_t = (p_{1,t}, \ldots, p_{n,t})$ et $p_{t+1} = (p_{1,t+1}, \ldots, p_{n,t+1})$ sont respectivement les probabilités de transition vers l'état X_t et X_{t+1}. Si nous considérons la présence d'un grand nombre de marcheurs sur le réseau, l'état au temps t représente la position de l'ensemble des marcheurs à cet instant particulier. Dès lors, la probabilité énoncée précédemment (2.14) vérifie une propriété intéressante. Effectivement,

$$\sum_{i=1}^{n} p_{i,t+1} \ = \ \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{A_{ji}}{k_j} p_{j,t} \ = \ \sum_{j=1}^{n} \frac{k_j}{k_j} p_{j,t} \ = \ \sum_{j=1}^{n} p_{j,t}.$$

Autrement dit, le nombre total de marcheurs aléatoires sur le réseau est conservé. Remarquons en sus que la probabilité de transition vers un certain état p_t est indépendante du temps t. Aussi le processus de marche aléatoire sur un réseau définit-il une chaîne de Markov stationnaire. La probabilité va donc tendre vers une distribution d'équilibre, donnée par

$$p^* = \frac{d^T}{2m}, \quad \text{ou encore,} \quad p_i^* = \frac{k_i}{2m} \tag{2.15}$$

pour i allant de 1 jusque n. Ceci correspond bien à un équilibre car si nous remplaçons $p_{i,t}$ par p_i^* dans la définition de la probabilité (2.14), nous obtenons l'égalité suivante,

$$p_{i,t+1} = \sum_{j=1}^{n} \frac{k_j}{2m} \frac{A_{ji}}{k_j}.$$

En utilisant la propriété 1.1 en page 7, nous pouvons transformer l'équation précédente comme suit,

$$p_{i,t+1} = \frac{k_i}{2m}.$$

Ceci revient à p_i^*, validant ainsi son statut de probabilité d'équilibre. De plus, celle-ci est unique car il s'agit d'un vecteur propre de la matrice T pour la valeur propre $\lambda = 1$ (en effet, $p^* = p^*T$) et ses éléments sont tous positifs. Or, comme énoncé dans les rappels théoriques en section 1.4, le théorème de Perron-Frobenius implique l'unicité d'un vecteur

3. La probabilité au temps $t + 1$ ne dépendant que de l'état au temps t

propre d'éléments tous positifs pour une matrice primitive. De plus, ce dernier est associé à la plus grande valeur propre de celle-ci. La matrice de transition T vérifie bien les hypothèses de ce théorème, la matrice d'adjacence A étant considérée primitive pour un réseau non-dirigé et connexe. Par la suite, nous utiliserons également la matrice $P^* = \text{diag}(p^*)$.

À présent, considérons que le réseau a été partagé en q communautés et, comme à l'accoutumée, nous notons g_i le groupe auquel appartient le sommet i. Les renseignements sur celles-ci peuvent être stockés dans une matrice $n \times q$, notée G, telle que les éléments de cette dernière, appelés G_{ij}, valent 1 si le sommet i appartient à la communauté g_j et 0 dans le cas contraire. Notre processus de Markov peut être observé pour cette partition en donnant un label, dénoté par un réel α_i, aux sommets de chacune des q communautés. Le processus observé, ou encore l'ensemble des états X_t, $(X_t)_{t\in\mathbb{N}}$, correspond alors à une suite de α_i.

Une bonne partition du réseau sur une certaine échelle de temps serait telle que l'état a de plus grandes chances de rester dans la communauté de départ sur cette période. La matrice $R_t(G)$ est donc définie comme ci-après par [Delvenne et al., 2009] pour obtenir ensuite la notion de stabilité du réseau au temps t. Ainsi,

$$R_t(G) = G^T \left(P^* T^t - {p^*}^T p^* \right) G. \tag{2.16}$$

Dès lors, « le premier terme de chaque élément $[R_t(G)]_{ij}$ correspond à la probabilité de démarrer la marche aléatoire dans la communauté g_i et de se trouver dans g_j après t étapes. À ce terme, nous retirons la probabilité que deux marcheurs aléatoires indépendants se trouvent respectivement dans les groupes g_i et g_j » ([Delvenne et al., 2009]). De cette manière, la stabilité met en lumière une comparaison entre la probabilité pour un marcheur de voyager d'un groupe de sommets à un autre et celle de la situation nulle de deux marcheurs indépendants.

De façon à atteindre une grande stabilité, les éléments diagonaux de $R_t(G)$ se doivent d'être plus grands que les autres. Effectivement, ils correspondent à la probabilité qu'une marche aléatoire commence et se termine dans la même communauté. Mais que se passe-t-il aux temps précédant t ? Il se pourrait que la marche aléatoire ait quitté la communauté pour y revenir, comme c'est le cas, par exemple, d'un graphe biparti [4]. Dans ce cas, la stabilité d'une telle marche doit être faible. Cette réflexion nous mène à la définition de la stabilité, notée r, à savoir

$$r_t(G) = \min_{0\leq s\leq t} \sum_{i=1}^{q} [R_s(G)]_{ii}. \tag{2.17}$$

Ce minimum nous assure que la partition ne pu être mauvaise lors des temps précédents. Ainsi, une partition vérifiant le cas décrit ci-avant se verra attribuer une faible stabilité.

4. « Un graphe est biparti s'il existe une partition des sommets en deux ensembles V_1 et V_2 telle que les sommets du premier ne sont adjacents qu'aux sommets du second et vice-versa. »[Lambiotte & Tabourier, 2013-2014]

Il convient d'analyser ensuite, pour chaque temps t, la division en communautés présentant la plus grande stabilité, c'est-à-dire analyser

$$r(t) = \max_{G} \ r_t(G).$$

Ainsi, les partitions les plus pertinentes seront celles offrant une grande stabilité sur une longue période car elles correspondent à une division du réseau valide sur plusieurs échelles de temps.

2.3.2 Méthode spectrale

La stabilité étant maintenant correctement définie, nous pouvons entamer la recherche d'un lien avec une méthode spectrale. Pour ce faire, nous allons, comme à la section 2.2, établir une équivalence avec la modularité. [Delvenne et al., 2009] et [Lambiotte et al., 2015] nous guident respectivement vers un lien avec la modularité simple (2.3, page 14) et la modularité généralisée (2.4, page 14).

Vers la modularité

Pour établir une connexion avec la modularité simple, nous nous basons, comme mentionné au début de cette sous-section, sur la source [Delvenne et al., 2009]. Il y est stipulé que la modularité est retrouvée lorsque le temps t vaut 1. Nous allons donc en établir la preuve.

Reprenons la définition de la matrice $R_t(G)$ (2.16) et adaptons la pour $t = 1$. Dès lors, nous obtenons l'égalité suivante,

$$R_1(G) = G^T \left(P^* T^1 - {p^*}^T p^* \right) G$$

ou encore, composante par composante,

$$[R_1(G)]_{ij} = \sum_{r=1}^{n} \left(\sum_{s=1}^{n} G_{is} (P^*_{ss} T_{sr} - p^*_s p^*_r) \right) G_{jr}.$$

En remplaçant les probabilités stationnaires et les matrices P^* et T par leur définition respective précédemment vue, nous avons que

$$[R_1(G)]_{ij} = \sum_{r=1}^{n} \left(\sum_{s=1}^{n} G_{is} \left(\frac{k_s}{2m} \frac{A_{sr}}{k_s} - \frac{k_s}{2m} \frac{k_r}{2m} \right) \right) G_{jr}.$$

En simplifiant le premier terme de la différence et en mettant en évidence le facteur $\frac{1}{2m}$, nous pouvons transformer l'équation précédente de la façon suivante,

$$[R_1(G)]_{ij} = \frac{1}{2m} \sum_{r=1}^{n} \left(\sum_{s=1}^{n} G_{is} \left(A_{sr} - \frac{k_s k_r}{2m} \right) \right) G_{jr}.$$

Par définition de la matrice G, l'élément G_{is} vaut zéro sauf si le sommet i appartient à la communauté g_s, ou encore, avec nos notations usuelles, si g_i est égale à g_s. Ainsi, nous

pouvons réécrire $[R_1(G)]_{ij}$ comme suit,

$$[R_1(G)]_{ij} = \frac{1}{2m}\sum_{r=1}^{n}\left(\sum_{s=1}^{n}\delta(g_i,g_s)\left(A_{sr}-\frac{k_sk_r}{2m}\right)\right)\delta(g_j,g_r).$$

En distribuant le deuxième symbole de Kronecker dans la somme interne, nous pouvons regrouper les sommes et ainsi obtenir la relation suivante,

$$[R_1(G)]_{ij} = \frac{1}{2m}\sum_{sr}\left(A_{sr}-\frac{k_sk_r}{2m}\right)\delta(g_i,g_s)\delta(g_j,g_r).$$

Or, la stabilité $r_1(G)$ étant définie par la trace de la matrice R_1, seuls les éléments diagonaux de celle-ci nous intéressent. Aussi obtenons-nous comme définition pour la trace de R_1 l'expression suivante,

$$\text{trace}(R_1(G)) = \sum_{i=1}^{q}\left(\frac{1}{2m}\sum_{sr}\left(A_{sr}-\frac{k_sk_r}{2m}\right)\delta(g_i,g_s)\delta(g_i,g_r)\right).$$

Par une propriété du symbole de Kronecker, à savoir $\sum_j \delta_{ij}\delta_{jk} = \delta_{ik}$, nous pouvons transformer la relation précédente de la manière suivante,

$$\text{trace}(R_1(G)) = \frac{1}{2m}\sum_{sr}\left(A_{sr}-\frac{k_sk_r}{2m}\right)\delta(g_s,g_r).$$

Nous arrivons finalement à une expression équivalente à la modularité (2.3, page 14) définie à la section 2.1. Conséquemment, maximiser $r(1)$ est équivalent à optimiser la modularité.

Vers la modularité généralisée

Afin de retrouver la modularité généralisée, nous suivrons [Lambiotte et al., 2015] révélant qu'il nous faut basculer en temps continu. Jusqu'à présent, les marcheurs aléatoires se déplaçaient tous en même temps et les probabilités de déplacement étaient exprimées en temps discret. Pour passer en temps continu, ces marcheurs bougeront avec un certain taux de probabilité. Par conséquent, en chaque sommet de réseau, nous assignons un processus de Poisson, celui-ci décrivant la probabilité qu'un certain nombre d'évènements se produisent. En effet, à chaque étape, chacun des marcheurs peut réaliser un certain nombre de sauts, cela correspond donc bien à une loi de Poisson et nous la supposerons identiquement distribuée parmi tous les sommets.

La dynamique de variation sur le réseau est donnée par

$$\frac{dp}{dt} = Tp - Ip,$$

où, comme précédemment, la matrice de transition $T = D^{-1}A$ et p correspond au vecteur de probabilité de transition. Par la mise en évidence du facteur $-p$, nous obtenons que

$$\frac{dp}{dt} = -p\left(I - D^{-1}A\right). \tag{2.18}$$

Nous avons vu, en section 1.4, la définition de la matrice Laplacienne, à savoir $L = D - A$. Aussi pouvons-nous l'utiliser pour simplifier l'expression 2.18 comme suit,

$$\frac{dp}{dt} = -pD^{-1}L. \tag{2.19}$$

En résolvant l'équation 2.19, nous obtenons l'expression suivante pour la valeur de la probabilité de transition p,

$$p = e^{-tD^{-1}L}p_0.$$

Celle-ci tend vers la même distribution d'équilibre que dans le cas discret, à savoir $p^* = \frac{d^T}{2m}$. Cela se vérifie aisément en substituant p par p^* dans l'équation 2.18. Dès lors, nous obtenons, pour i allant de 1 à n, la relation suivante,

$$\frac{dp_i}{dt} = -\frac{k_i}{2m} + \sum_{l=1}^{n} \frac{k_l}{2m}\frac{A_{li}}{k_l}.$$

Par la relation liant la matrice d'adjacence aux degrés des sommets (1.1, page 7), nous pouvons transformer l'équation précédente comme suit,

$$\frac{dp_i}{dt} = -\frac{k_i}{2m} + \frac{k_i}{2m},$$

annulant ainsi la dérivée première de la probabilité de transition. Conséquemment, p^* correspond bien à un équilibre. Par la suite nous considérerons également $P^* = \text{diag}(p^*)$.

La stabilité de Markov est définie dans la source [Lambiotte et al., 2015] pour un temps continu par

$$r_t(G) = \text{trace}\left[G^T\left(P^*e^{-tD^{-1}L} - {p^*}^T p^*\right)G\right]. \tag{2.20}$$

Comme énoncé au début de cette section, nous allons travailler cette expression de manière à retrouver la modularité généralisé définie précédemment par l'équation 2.4 en page 14. Dans cet ordre d'idées, nous effectuons un développement linéaire de l'exponentielle limité au premier ordre. Dès lors, nous obtenons que

$$e^{-tD^{-1}L} \;=\; e^{-0D^{-1}L} + (-tD^{-1}L)'e^{-0D^{-1}L}(t-0) \;=\; 1 - D^{-1}Lt.$$

En utilisant cette expression dans l'équation 2.20, nous pouvons réécrire la définition de la stabilité de la façon suivante,

$$r_t(G) = \text{trace}\left[G^T\left(P^* - P^*D^{-1}Lt - {p^*}^T p^*\right)G\right]. \tag{2.21}$$

Notons par $R_t(G)$ la partie interne de l'équation 2.21, autrement dit, la matrice sur laquelle la trace est calculée. Aussi, en nous souvenant que $L = D - A$, chacune des composantes de cette matrice se définit par

$$[R_t(G)]_{ij} = \sum_{r=1}^{n}\left(\sum_{s=1}^{n}\delta(g_i,g_s)\left(\frac{k_s}{2m} - \frac{k_s}{2m}\frac{1}{k_s}(k_s - A_{sr})t - \frac{k_s}{2m}\frac{k_r}{2m}\right)\delta(g_r,g_j)\right)$$

Or, par définition de $r_t(G)$, seuls les éléments diagonaux de la matrice $R_t(G)$ seront considérés. Par voie de conséquence, en utilisant une propriété du symbole de Kronecker précédemment rencontrée, la stabilité devient

$$r_t(G) = \sum_{rs} \frac{k_s}{2m}\delta(g_r, g_s) - t\sum_{rs} \frac{k_s}{2m}\delta(g_r, g_s) + t\frac{1}{2m}\sum_{rs} A_{rs}\delta_{rs} - \frac{1}{2m}\sum_{rs} \frac{k_s k_r}{2m}\delta(g_r, g_s),$$

ou encore, en rassemblant les termes,

$$r_t(G) = (1-t)\frac{1}{2m}\sum_{rs} k_s\delta(g_r, g_s) + \frac{1}{2m}\sum_{rs}\left(tA_{rs} - \frac{k_s k_r}{2m}\right)\delta(g_r, g_s).$$

En développant le premier terme dont le symbole de Kronecker peut être agrégé dans la somme, nous obtenons l'expression suivante,

$$r_t(G) = (1-t)\frac{1}{2m}\sum_{s} k_s + \frac{1}{2m}\sum_{rs}\left(tA_{rs} - \frac{k_s k_r}{2m}\right)\delta(g_r, g_s).$$

Or, par la propriété 1.2 en page 7, la somme des degrés d'un graphe équivaut au double du nombre total de ses arêtes. Conséquemment, le premier terme se réduit à $(1-t)$. En outre, l'optimisation de la stabilité s'effectuant par rapport à la répartition des sommets en communautés, il est considéré comme une constante et peut donc être oublié. Nous sommes finalement revenu à la modularité généralisée définie en page 14 par l'équation 2.4 pour une constante γ valant $1/t$.

2.4 Conclusion

Ce chapitre portait sur l'analyse théorique des différentes méthodes précédemment introduites en section 1.5. Celles-ci abordaient chacune d'une manière spécifique le problème de détection de communautés, à savoir via l'optimisation de la modularité, l'inférence statistique et l'optimisation de la stabilité. Dans leurs sections respectives, elles ont été formellement définies puis développées en vue de trouver leur point commun. Ce dernier s'est révélé être la modularité généralisée,

$$Q(\gamma) = \frac{1}{2m}\sum_{ij}\left(A_{ij} - \gamma\frac{k_i k_j}{2m}\right)\delta(g_i, g_j).$$

Effectivement, la méthode d'inférence statistique pour le modèle de partition implantée équivaut à l'optimisation de cette mesure lorsque

$$\gamma = \frac{(\omega_{\text{in}} - \omega_{\text{out}})}{(\log\omega_{\text{in}} - \log\omega_{\text{out}})}.$$

La stabilité préalablement linéarisée, quant à elle, est analogue à la modularité généralisée pour le paramètre $\gamma = 1/t$. Néanmoins, avoir obtenu ces équivalences ne suffit pas. Effectivement, notre objectif initial était d'obtenir une méthode spectrale pour toutes les méthodes. Or, en section 2.1.2, nous avons abouti à une telle méthode pour l'optimisation de la modularité généralisée. Les équivalences susmentionnées nous permettent alors d'atteindre la spectralité de l'ensemble des méthodes considérées dans ce mémoire.

Implémentation et application des méthodes sur des réseaux réels

Les différentes méthodes ayant été analysées de manière théorique, nous nous intéressons à leur application sur des réseaux concrets. En effet, leur équivalence a été prouvée théoriquement, mais qu'en est-il en pratique ? Dans ce chapitre, nous tenterons de répondre à cette question.

En premier lieu, nous présenterons les codes utilisés pour l'implémentation des trois méthodes ainsi que les réseaux sur lesquels nous appliquerons ces dernières. Nous poursuivrons ensuite par diverses analyses et comparaison des résultats obtenus.

3.1 Présentation des codes

Les codes utilisés pour l'application concrète de la méthode d'optimisation de la stabilité proviennent de [Schaub et al., 2015]. Ils sont le fruit de l'implémentation de la méthode détaillée dans ce mémoire et dans [Delvenne et al., 2009]. La fonction « stability » dont l'entête est la suivante, est utilisée pour optimiser la stabilité et trouver ainsi la meilleure partition du réseau pour un temps donné.

```
function [S, N, VI, C] = stability(G, T, varargin)
%STABILITY    Graph partitioning optimizing stability with the Louvain
%             algorithm
%
%   [S, N, VI, C] = STABILITY(G, T) finds the optimal partitions of the
%   graph G by optimizing the stability at each Markov time in vector T. G
%   can either be the list of the edges in the graph (in the form [node i,
%   node j, weight of link i-j; node k, node l, weight of link k-l;...] if
%   the graph is weighted, or [node i, node j; node k, node l;...] if the
%   graph is unweighted) or the adjacendy matrix of the graph. S, N, VI and
%   C contain respectively the stability, the number of communities, the
%   variation of information, and the optimal partition for each
%   Markov time contained in T. If T is not specified, the modularity
%   (equivalent to stability for T=1) is calculated. Ideally, Markov time
%   should be sampled exponentially (e.g.: T = 10.^[-2:0.01:2]).
%   [S, N, VI, C] = STABILITY(G, T,'PARAM',VALUE) accepts one or more
%   comma-separated parameter name/value pairs.
```

La stabilité ayant été prouvée équivalente à la modularité généralisée en section 2.3 pour le paramètre de résolution $\gamma = 1/t$ où t représente le temps, ce même code correspond également à l'optimisation de la modularité. Conséquemment, dans la suite de ce chapitre, nous ne distinguerons plus les méthodes liées à la stabilité et à la modularité, leur algorithme étant identique.

En ce qui a trait à l'inférence statistique sur le modèle de partition implantée, ce code ne peut être utilisé sans modification préalable. En effet, nous avions établi l'équivalence entre cette méthode et la modularité généralisée pour une valeur du paramètre de résolution défini comme suit,

$$\gamma = \frac{(\omega_{\text{in}} - \omega_{\text{out}})}{(\log \omega_{\text{in}} - \log \omega_{\text{out}})}.$$

Pour rappel, dans le modèle de partition implantée, les arêtes sont placées selon une probabilité de Poisson de moyenne ω_{in} si les deux sommets liés à l'arête considérée sont de la même communauté et ω_{out} dans le cas contraire. En fin de section 2.2.2, nous avions formulé le problème de non connaissance de ces deux paramètres avant le calcul de la modularité et donc de γ. Cependant, [Newman, 2016] proposait une méthode itérative pour palier à ce problème. Ainsi, une estimation originelle du paramètre γ est utilisée pour calcul initial de la modularité, nous menant de cette façon à une première partition du réseau. De là, nous pouvons obtenir les valeurs des paramètre ω_{in} et ω_{out} par les deux relations suivantes données par [Newman, 2016],

$$\omega_{\text{in}} = \frac{2m_{\text{in}}}{\sum_r \frac{\kappa_r^2}{2m}} \quad \text{et} \quad \omega_{\text{out}} = \frac{2m - 2m_{in}}{2m - \sum_r \frac{\kappa_r^2}{2m}},$$

où m_{in} et m_{out} représentent respectivement le nombre total d'arêtes à l'intérieur des groupes et entre ceux-ci. L'élément κ_r, quant à lui, correspond à la somme des degrés des sommets du groupe r. En utilisant ces deux valeurs de ω_{in} et ω_{out}, nous pouvons obtenir une nouvelle estimation de γ et répéter ensuite l'ensemble du processus dans l'espoir d'atteindre une convergence. Néanmoins, notre source stipule que celle-ci n'est pas garantie pour les réseaux réels. Les codes présentant les modifications susmentionnées se trouvent en Annexe B.

En ce qui a trait à ces derniers, nous utiliserons les deux réseaux fournis avec les codes par [Schaub et al., 2015], à savoir « Protein Adk » et « Ring of rings ». Nous ferons également usage des graphes décrivant les amitiés existant entre les membres du club de karaté à la US university dépeintes dans [Zachari, 1977], les relations au sein d'une communauté de dauphins vivant en Nouvelle-Zélande étudiées par [Lusseau et al., 2003], les personnages du livre de Victor Hugo « Les misérables » apparaissant dans un même chapitre retracés dans [Knuth, 1993] et, finalement, les noms et adjectifs utilisés de concert dans le livre « David Copperfield » de Charles Dickens détaillés par [Newman, 2006]. Le lecteur intéressé peut trouver une vue de ces six réseaux en Annexe A.

3.2 Application des deux méthodes implémentées aux six réseaux

L'analyse des divers résultats obtenus par les algorithmes se fera en trois parties. En premier lieu, nous regarderons l'évolution du nombre de communautés et de la valeur de modularité généralisée en fonction du temps t et donc du paramètre de résolution γ. En second lieu, la robustesse des deux algorithmes sera analysée. Finalement, la variation d'information sera utilisée pour comparer les partitions produites par les deux méthodes.

3.2.1 Analyses des résultats sur le nombre de communautés et la modularité

Pour commencer, analysons les résultats issus de l'application concrète de l'algorithme lié à l'optimisation de la stabilité. En FIGURES 3.1 et 3.2, une décroissance du nombre de communautés et de la valeur de la stabilité en fonction de la croissance du paramètre associé au temps t est observable pour les six réseaux. Le paramètre dit « de résolution », γ, valant l'inverse de ce dernier, cette décroissance s'opérerait pour des valeurs de γ s'amenuisant. L'algorithme aurait tendance à identifier de grandes communautés peu nombreuses pour de grandes valeurs du temps et, en opposition, une kyrielle de petits groupes de sommets

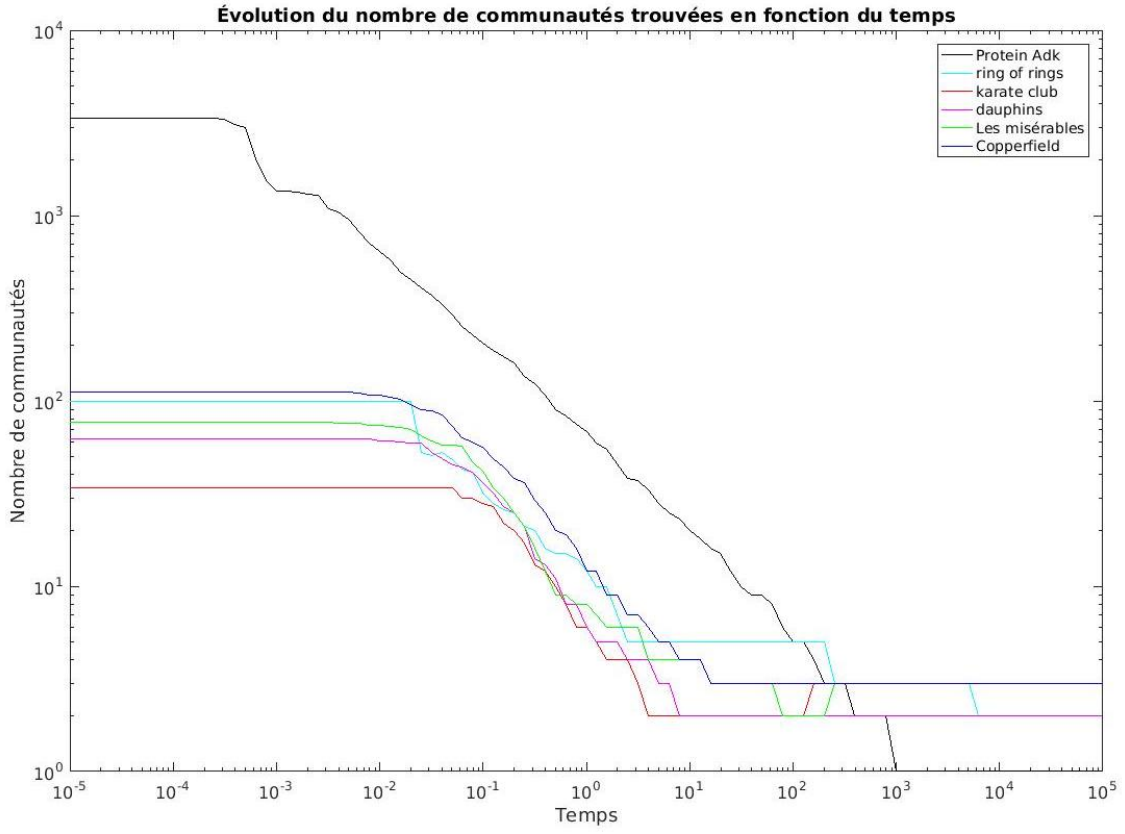

FIGURE 3.1: Évolution du nombre de communautés obtenues par l'algorithme lié à la stabilité en fonction du temps.

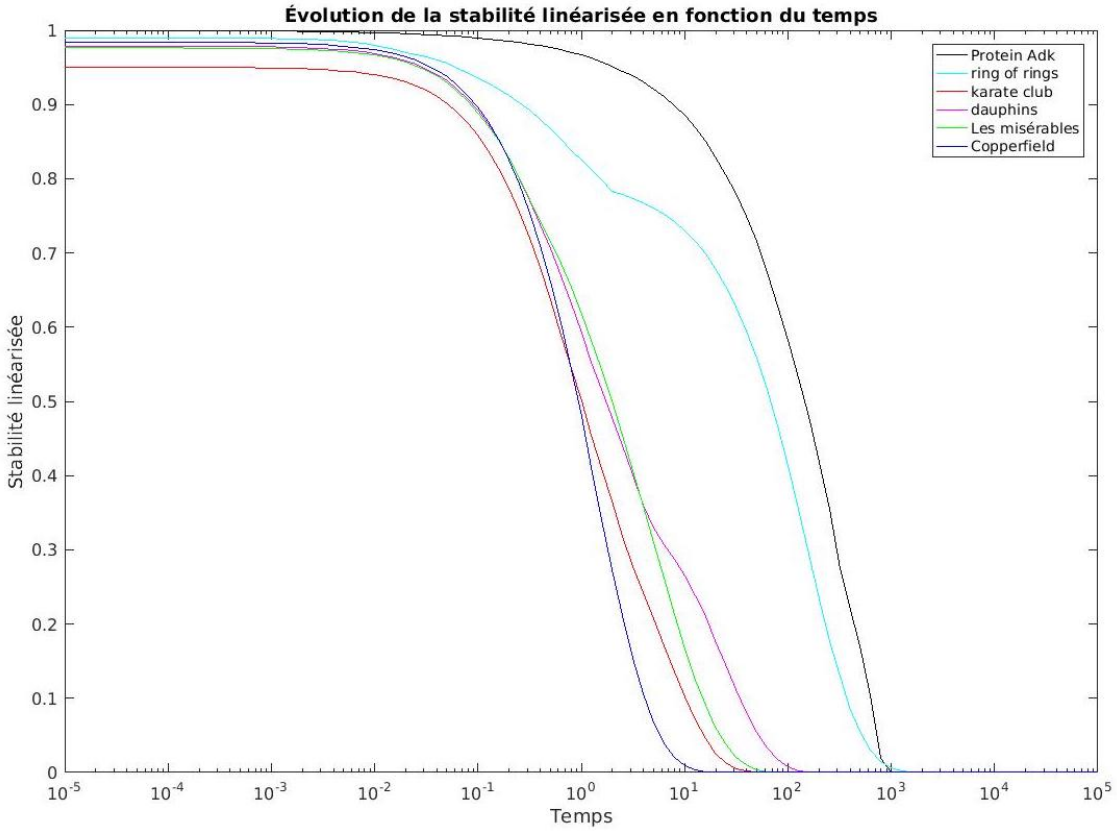

FIGURE 3.2: Évolution de la stabilité calculée par l'algorithme lié à la stabilité en fonction du temps.

pour des valeurs faibles de ce paramètre. En effet, si nous nous référons à la définition de la modularité généralisée rappelée ci-après pour $\gamma = 1/t$,

$$Q(\gamma) = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \gamma \frac{k_i k_j}{2m} \right) \delta(g_i, g_j),$$

augmenter la valeur de t (et donc diminuer celle de γ) accorderait plus d'importance au premier terme de la somme, maximisant par conséquent la modularité lorsque de nombreux sommets se retrouvent dans la même communauté. À l'inverse, un paramètre t tendant vers zéro est associé au cas de figure où chaque sommet appartient à sa propre communauté. En effet, comme nous pouvons le voir en FIGURE 3.1, les valeurs atteintes par les diverses courbes pour un temps valant 10^{-5} correspond au nombre de sommets respectif des différents réseaux. Finalement, lorsque t tend vers zéro, la stabilité s'approche de l'unité, la valeur maximale de cette mesure. Cette dernière observation s'explique en reprenant la définition de la stabilité (2.20, en page 28) pour $t = 0$. Nous obtenons la définition suivante,

$$r_0(G) = \text{trace} \left[G^T \left(P^* - {p^*}^T p* \right) G \right].$$

Notons $R_0(G)$ la matrice sur laquelle la trace est calculée, chacune de ses composantes est définie par

$$[R_0(G)]_{ij} = \sum_{r=1}^{n} \left(\sum_{s=1}^{n} \delta(g_i, g_s) \left(\frac{k_s}{2m} - \frac{k_s k_r}{4m^2} \right) \delta(g_r, g_j) \right).$$

Pour obtenir la stabilité en $t = 0$, seuls les éléments diagonaux de cette matrice seront pris en compte. Par voie de conséquence, nous obtenons que

$$r_0(G) = \sum_{i=1}^{n} \left(\sum_{rs} \left(\frac{k_s}{2m} - \frac{k_s k_r}{4m^2} \right) \delta(g_i, g_s) \delta(g_r, g_i) \right).$$

Le symbole de Kronecker vérifiant la propriété suivante, $\sum_i \delta_{is}\delta_{ri} = \delta_{rs}$, nous pouvons réécrire l'équation précédente comme suit,

$$r_0(G) = \sum_{ij} \left(\frac{k_s}{2m} - \frac{k_s k_r}{4m^2} \right) \delta(g_i, g_j).$$

En séparant ceci en deux sommes et en utilisant la propriété (1.2, en page 7) à présent bien connue, nous arrivons à la relation suivante,

$$r_0(G) = \frac{2m}{2m} - \sum_{ij} \frac{k_i k_j}{4m^2} \delta(g_i, g_j).$$

En développant davantage chacun des termes du membre de droite, nous obtenons finalement le résultats suivant,

$$r_0(G) = 1 - \sum_{i} \frac{k_i^2}{4m^2}.$$

Le nombre d'arêtes total d'un réseau réel étant généralement important et le deuxième terme étant inversement proportionnel à m^2, celui-ci tend vers zéro de manière quadratique. Conséquemment, un temps nul engendre une valeur de stabilité proche l'unité, expliquant ainsi l'allure de la FIGURE 3.2.

Pour ce qui est de la méthode inférentielle, une valeur initiale du temps est donnée à l'algorithme qui la recalcule ensuite à chaque itération selon le principe détaillé en section 3.1. Ceci a lieu jusqu'à l'obtention d'une convergence atteinte si deux valeurs de la stabilité diffèrent de moins de 10^{-6}. En FIGURES 3.3 et 3.4, il est à remarquer une disparition de la tendance à la décroissance observée dans les résultats précédents. A contrario, comme le montre le deuxième graphique, la modularité généralisée semble meilleure pour des valeurs initiales variant entre 10^{-1} et l'unité [1]. Elle n'accède cependant pas à un niveau plus haut que 0.5 pour la plupart de nos réseaux, à l'exception de « Protein Adk » et « Ring of rings ».

La FIGURE 3.5, quant à elle, montrant la valeur optimale obtenue pour le paramètre γ, semble indiquer une tendance à croître en fonction de la valeur initiale de celui-ci. Aussi,

1. L'article [Newman, 2016] conseillait une valeur initiale de γ égale à l'unité.

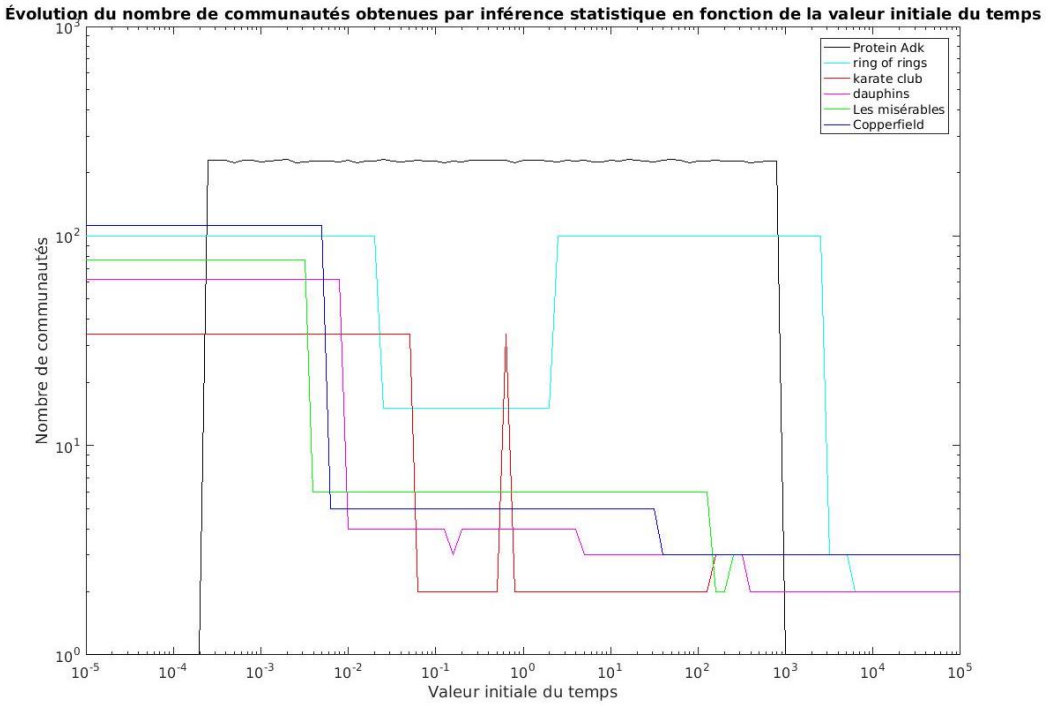

FIGURE 3.3: Évolution du nombre de communautés obtenues par l'algorithme lié à l'inférence en fonction de la valeur initiale du temps.

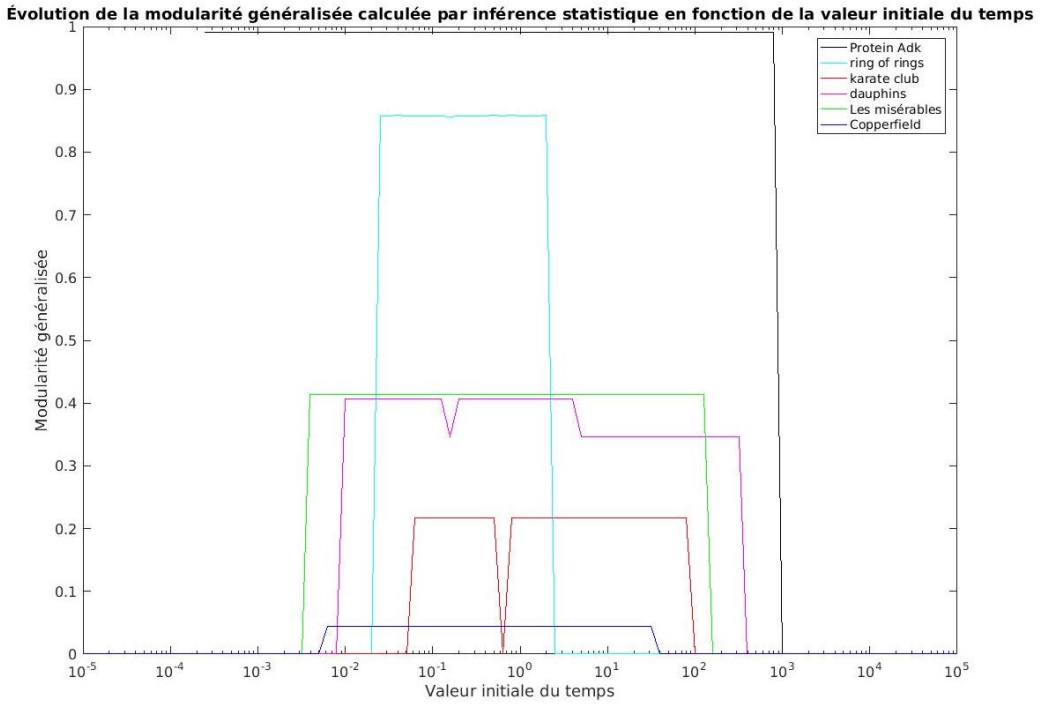

FIGURE 3.4: Évolution de la stabilité calculée par l'algorithme lié à l'inférence en fonction de la valeur initiale du temps.

à l'exception de la courbe liée au réseau décrivant la protéine Adk, des valeurs initiales du paramètre de résolution trop petites donnent à celui-ci une valeur finale trop proche de zéro. Enfin, nous avons vu que les valeurs maximales, néanmoins peu élevées, pour la modularité étaient atteintes pour une initialisation du temps entre 10^{-1} et l'unité. En se référant à cette figure, nous pouvons trouver les valeurs finales du temps obtenues pour cette fourchette. Celles-ci sont proche de 10^{-1} pour le réseau lié à la protéine Adk, aux environs de l'unité pour le graphe « Ring of rings » et à proximité de dix pour les quatre réseaux restants. Cela semble s'accorder avec les résultats obtenus pour la méthode liée à la stabilité. Effectivement, la FIGURE 3.2 montre des valeurs respectivement similaires pour la stabilité de ces six réseaux pour chacun des temps susmentionnés.

3.2.2 Robustesse des algorithmes

Au niveau de la robustesse des algorithmes, pour chaque valeur du paramètre lié au temps, une vingtaine d'itérations ont été effectuées et la variance des résultats obtenus a été calculée. Bien que l'algorithme lié à la stabilité nous donne une variance ne dépassant pas 10^{-5}, en FIGURES 3.6 et 3.7, nous observons une meilleure robustesse dans les résultats obtenus par la méthode d'inférence statistique d'une manière générale.

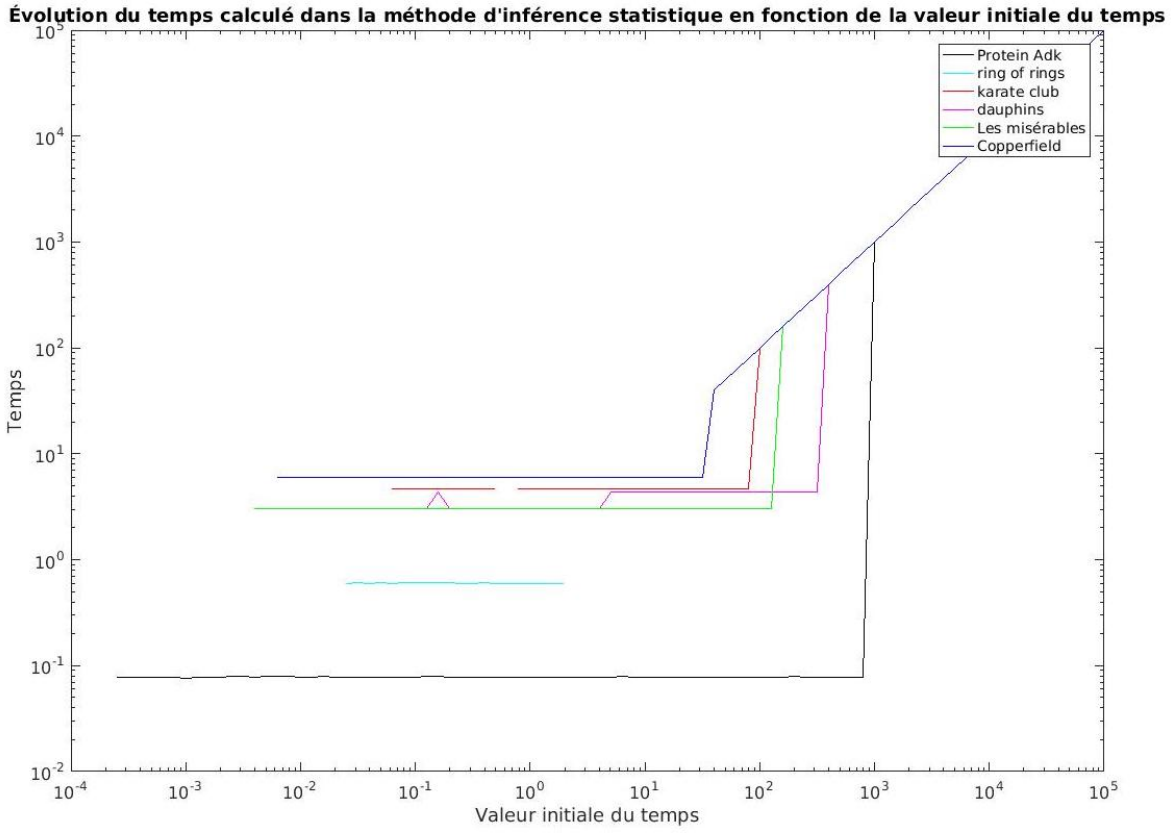

FIGURE 3.5: Évolution de la valeur optimale du paramètre de résolution calculée par l'algorithme lié à l'inférence en fonction de la valeur initiale du temps.

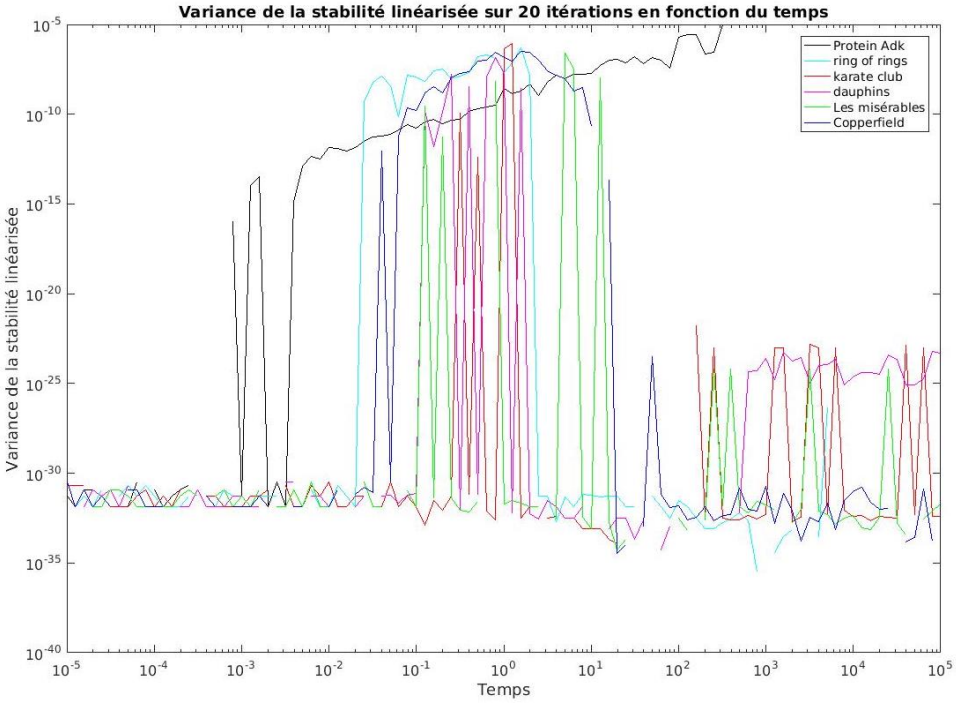

FIGURE 3.6: Variance de la stabilité calculée par l'algorithme lié à la stabilité sur vingt itérations en fonction du temps.

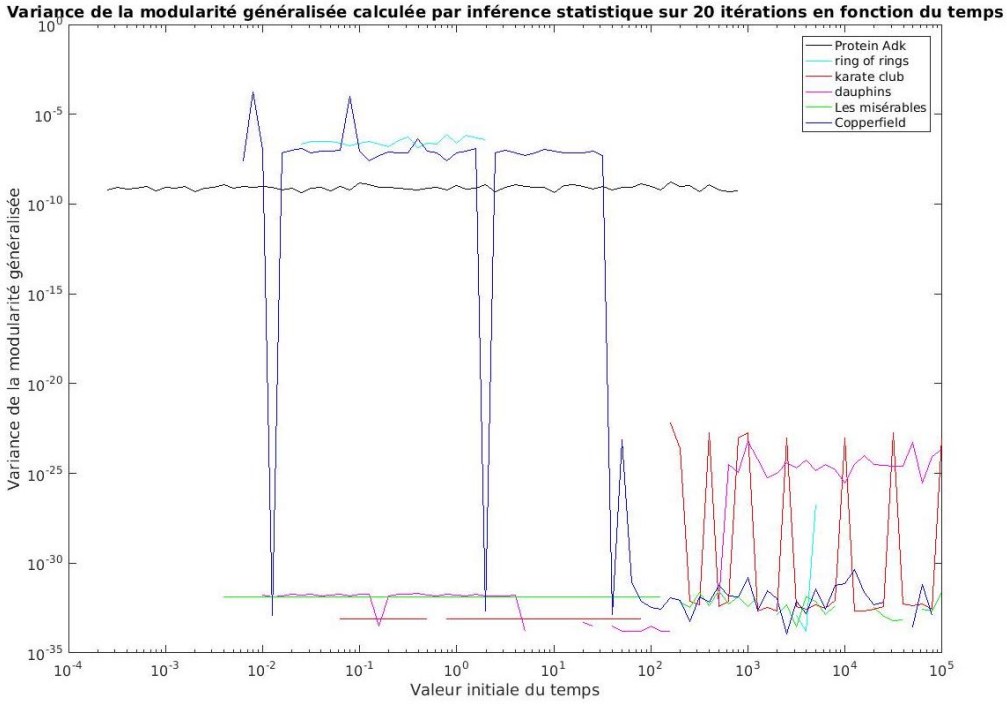

FIGURE 3.7: Variance de la modularité généralisée calculée par l'algorithme lié à l'inférence sur vingt itérations en fonction de la valeur initiale du temps.

3.2.3 Analyse de la variation d'information

La variation d'information est une mesure bien connue dans l'analyse de partitions. Elle détermine l'écart entre deux de ces dernières. Cela nous permet ainsi de comparer chacune des divisions d'un réseau obtenues par nos deux algorithmes sur la fourchette de temps précédemment utilisée en section 3.2.1. Nous analyserons donc les valeurs de cette mesure pour chacun de nos six réseaux.

L'ensemble des FIGURES reprenant ces résultats se trouvent en Annexe C. Nous pouvons observer la présence des plus faibles valeurs de la variation d'information lorsque le nombre de communautés produites par les deux méthodes correspondent. Par voie de conséquence, cela signifierait que les deux méthodes ont tendance à trouver la même partition de sommets lorsque le nombre de communautés concorde, validant ainsi l'équivalence théorique identifiée au chapitre précédent.

3.3 Conclusion

Ce chapitre visait à analyser l'application des trois méthodes précédemment vues sur six réseaux concrets. Nous avons utilisé les codes [Schaub et al., 2015] pour l'implémentation de l'approche liée à la stabilité. L'équivalence entre celle-ci et la modularité généralisée ayant été prouvée au chapitre précédent, ces codes correspondaient également à l'optimisation de la modularité. Enfin, pour implémenter la méthode liée à l'inférence statistique, quelques modifications définies en section 3.1 ont été apportées.

En ce qui a trait aux analyses elles-mêmes, nous avons constaté une décroissance du nombre de communautés trouvées et de la stabilité calculée par le premier algorithme en fonction du paramètre du temps. Cette tendance n'a en revanche pas été observée pour les résultats en lien avec la seconde méthode. Quant à la robustesse de ces deux algorithmes, bien qu'elle soit assez bonne pour chacun d'eux, elle semble meilleure pour le second. Pour terminer, quand les deux algorithmes trouvent à peu près le même nombre de communautés, la variation d'information est fort faible, confirmant ainsi l'équivalence entre toutes ces méthodes.

Conclusion

Dans ce mémoire, nous nous intéressions aux réseau et plus particulièrement à la détection de groupes de sommets, ou communautés, dans ces derniers. La pluralité des méthodes développées pour la résolution de ce problème nous a poussé à rechercher l'existence d'un point commun entre celles-ci.

La première partie apportait une définition formelle de chacune de ces méthodes et guidait le lecteur vers leur point commun. Sur base du paramètre de résolution γ, la modularité généralisée constitue le point de rencontre entre celles-ci. Effectivement, l'inférence statistique avec le modèle de partition implantée équivaut à l'optimisation de cette mesure lorsque

$$\gamma = \frac{(\omega_{\text{in}} - \omega_{\text{out}})}{(\log \omega_{\text{in}} - \log \omega_{\text{out}})}$$

et la stabilité préalablement linéarisée est analogue à la modularité généralisée pour le paramètre $\gamma = 1/t$. L'équivalence théorique a donc été prouvée et de part le développement en section 2.1.2 permet d'obtenir une version spectrale de l'ensemble de nos méthodes.

Dans la seconde partie, nous avons appliqué ces méthodes sur six réseaux particuliers et ce, pour vérifier ces équivalences d'un point de vue pratique. Il est apparu que, en fonction des valeurs initiales du paramètre de résolution, les algorithmes affichaient des tendances différentes quant à l'évolution de la modularité ou du nombre de communautés trouvées. Toutefois, elles se rejoignaient dans la répartition des sommets des réseaux dans celles-ci lorsque ces méthodes découvraient une quantité de collectivités similaire. Les algorithmes identifiant les mêmes communautés, l'équivalence leur équivalence d'un point de vue pratique est dès lors atteinte.

Bibliographie

[Brandes et al.,2008] Brandes, U., Delling, D., Gaertler, M., Gorke, R., Hoefer, R., Nikoloski, Z., Wagner, D., *On modularity clustering*, Knowledge and Data Engineering, IEEE Transactions on, vol. 20, no. 2, pp. 172–188, 2008.

[Clauset et al., 2008] Clauset, A., Moore, C., Newman, M.E.J., *Hierarchical structure and the prediction of missing links in networks*, Nature, Vol. 453, 98-101, doi :10.1038/nature06830, 2008.

[Clauset et al., 2008 (bis)] Clauset, A., Moore, C., Newman, M.E.J., *Hierarchical structure and the prediction of missing links in networks : Supplementary Information*, Nature, Vol. 453, 98-101, doi :10.1038/nature06830, 2008.

[Delvenne et al., 2009] Delvenne, JC., Yaliraki, S.N., Barahona, M., *Stability of graph communities across time scales*, arXiv :0812.1811 [physics.soc-ph], 2009.

[Fortunato & Barthélemy, 2006] Fortunato, S., Barthélemy, M., *Resolution limit in community detection*, doi : 10.1073/pnas.0605965104, 2006.

[Kashtan & Alon, 2005] Kashtan, N., Alon, U., *Spontaneous evolution of modularity and network motifs*, Proc Natl Acad Sci USA, Vol. 102, Num. 39,13773-13779, 2005.

[Knuth, 1993] Knuth, D.E., *The Stanford GraphBase : A Platform for Combinatorial Computing*, Addison-Wesley, Reading, MA, 1993.

[Lambiotte et al., 2015] Lambiotte, R., Delvenne, JC., Barahona, M., *Random Walks, Markov Processes and the Multiscale Modular Organization of Complex Networks*, arXiv :1502.04381 [physics.soc-ph], 2015.

[Lambiotte & Masuda, 2016] Lambiotte, R., Masuda, N., *A Guide to Temporal Networks*, Series on complexity science, Vol. 4, 2016.

[Lambiotte & Tabourier, 2013-2014] Lambiotte, R., Tabourier, L., *Théorie des graphes*, Syllabus, Université de Namur, 2013-2014.

[Lemaître, 2013-2014] Lemaître, A., *Algèbre linéaire : 1er baccalauréat en mathématique et physique*, Syllabus, Université de Namur, 2013-2014.

[Lusseau et al., 2003] Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E., Dawson, S.M., *The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations*, Behavioral Ecology and Sociobiology 54, 396-405, 2003.

[Meunier et al., 2010] Meunier, D., Lambiotte, R., Bullmore, E.T., *Modular and hierarchically modular organization of brain networks*, Front. Neurosci, Focused Review, doi.org/10.3389/fnins.2010.00200, 2010.

[Meyer, 2000] Meyer, C.D., *Matrix Analysis and Applied Linear Algebra*, SIAM, Philadelphia, 2000.

[Newman, 2006] Newman, M.E.J., *Finding community structure in networks using the eigenvectors of matrices*, Preprint physics/0605087, 2006.

[Newman, 2010] Newman, M.E.J., *Networks : An Introduction*, Oxford University Press Inc., New York, 2010.

[Newman, 2016] Newman, M.E.J., *Community detection in networks : Modularity optimization and maximum likelihood are equivalent*, arXiv :1606.02319 [cs.SI], 2016.

[Pan & Sinha, 2009] Pan, R.K., Sinha, S., *Modularity produces small-world networks with dynamical time-scale separation*, arXiv :0802.3671v2 [physics.bio-ph], 2009.

[Porter et al., 2009] Porter, M.A., Onnela, JP., Mucha, P.J., *Communities in Networks*, Notice of the AMS, Vol. 56, Num. 9, 1082-1166, 2009.

[Reichardt & Bornholdt, 2008] Reichardt, J., Bornholdt, S., *Statistical Mechanics of Community Detection*, arXiv :cond-mat/0603718v1 [cond-mat.dis-nn], 2008.

[Robinson et al., 2009] Robinson, P.A., Henderson, J.A., Matar, E., Riley, P., Gray, R.T., *Dynamical Reconnection and Stability Constraints on Cortical Network Architecture*, Physical Review Letters, DOI : 10.1103/PhysRevLett.103.108104, 2009.

[Rubinov et al., 2009] Rubinov, M., Sporns, O., Van Leeuwen, C., Breakspear, M., *Symbiotic relationship between brain structure and dynamics*, BMC Neuriscience, doi :10.1186/1471-2202-10-55, 2009.

[Schaub et al., 2015] Schaub, M., Yaliraki, S., Barahona, M., Delmotte, A., *PartitionStability*, GitHub repository, https ://github.com/michaelschaub/PartitionStability, 2015.

[Sporns et al., 2004] Sporns, O., Chialvo, D.R., Kaiser, M., Hilgetag, C.C., *Organization, development and function of complex brain networks*, TRENDS in Cognitive Sciences, Vol. 8, Num. 9, 2004.

[Zachari, 1977] Zachary, W.W., *An information flow model for conflict and fission in small groups*, Journal of Anthropological Research 33, 452-473, 1977.

Annexe A

Visualisation des six réseaux

Protéine Adk

Ce réseau était fourni avec les codes de [Schaub et al., 2015] et décrit les liens existant entre les acides aminés composant une protéine Adk. Ces caractéristiques sont les suivantes,

- ⋆ 3340 sommets ;
- ⋆ 4921 arêtes ;
- ⋆ pondéré ;
- ⋆ non dirigé.

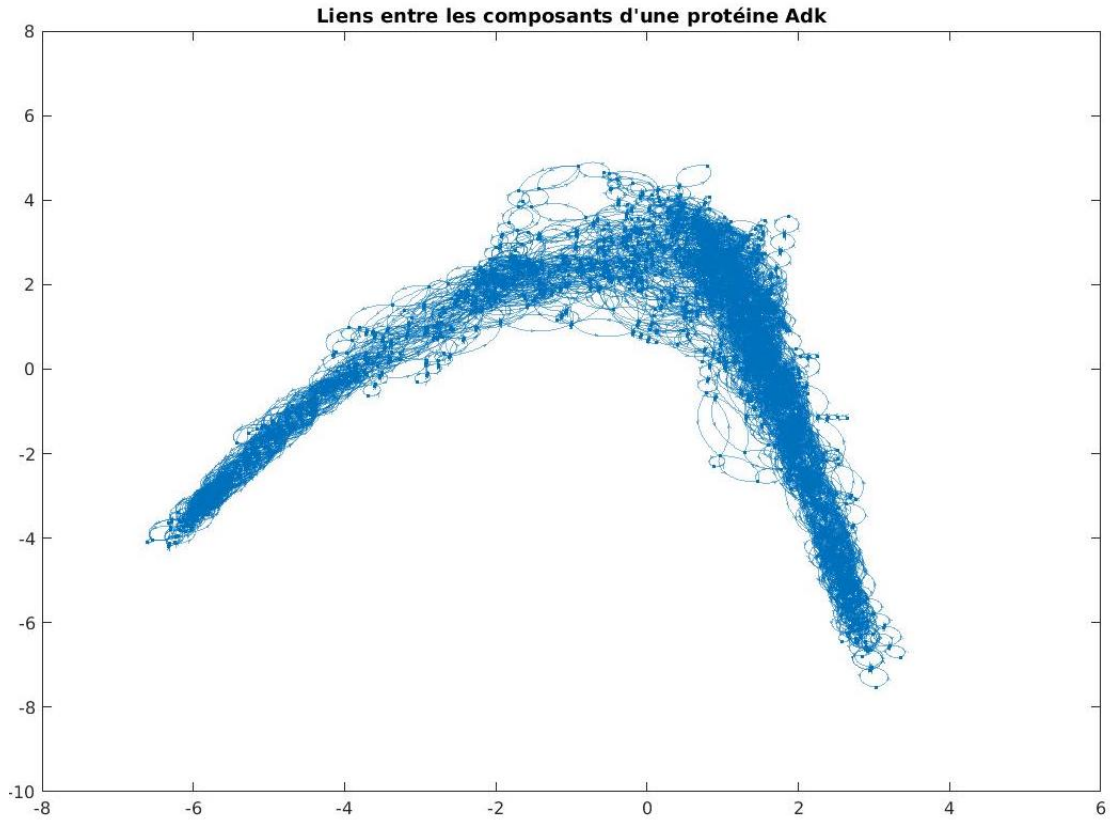

FIGURE A.1: Protéine Adk

Ring of rings

Ce réseau était également fourni avec les codes de [Schaub et al., 2015] et il possède les caractéristiques suivantes,

- ⋆ 100 sommets ;
- ⋆ 105 arêtes ;
- ⋆ pondéré ;
- ⋆ non dirigé.

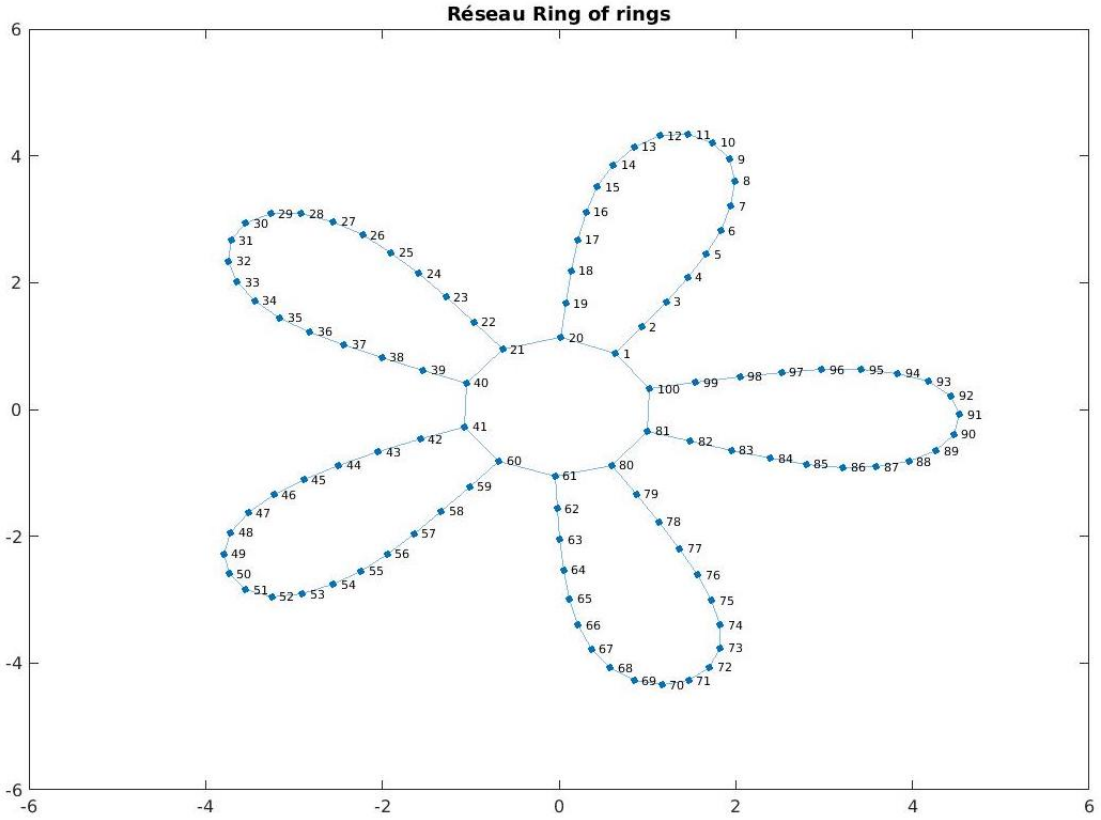

FIGURE A.2: Ring of rings

Club de karaté

Ce réseau décrit les liens d'amitié présents entre les membres du club de karaté de l'US university en 1977. Il possède les caractéristiques suivantes,

- ⋆ 34 sommets ;
- ⋆ 78 arêtes ;
- ⋆ non pondéré ;
- ⋆ non dirigé.

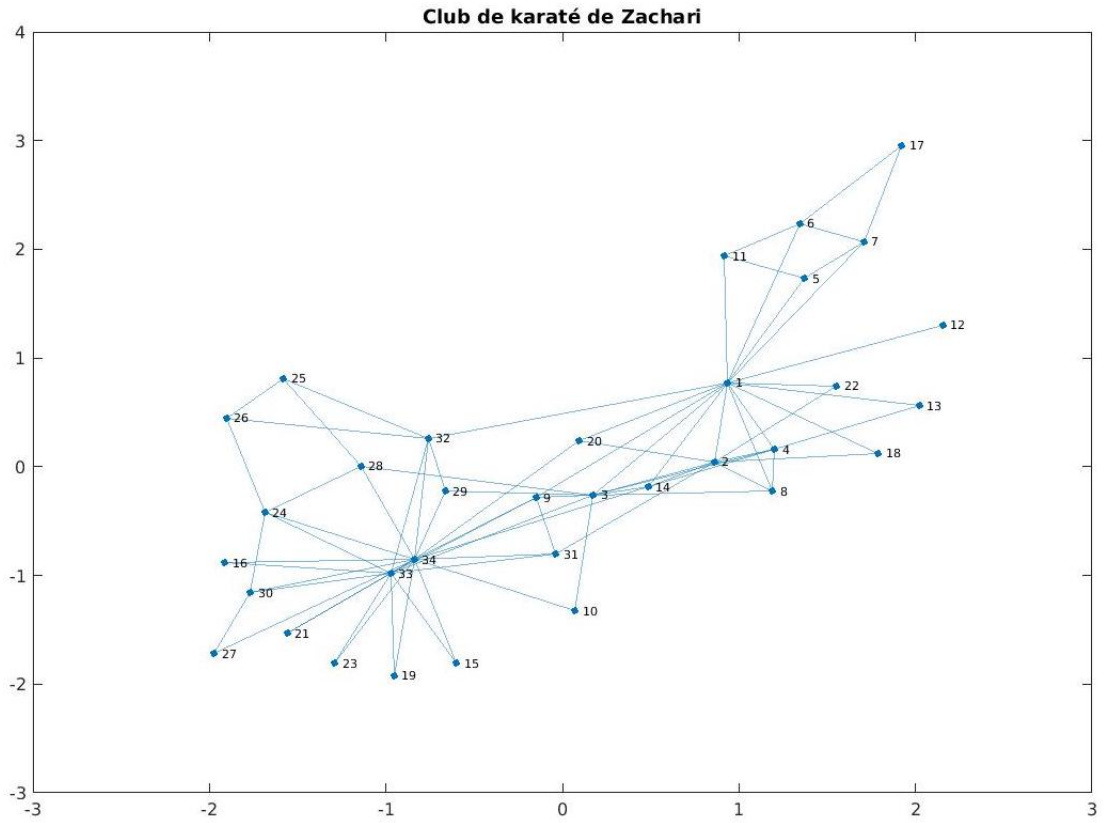

FIGURE A.3: Club de karaté

Dauphins

Ce réseau décrit les relations au sein d'une communauté de dauphins de Nouvelle-Zélande. Il possède les caractéristiques suivantes,

- ⋆ 62 sommets ;
- ⋆ 159 arêtes ;
- ⋆ non pondéré ;
- ⋆ non dirigé.

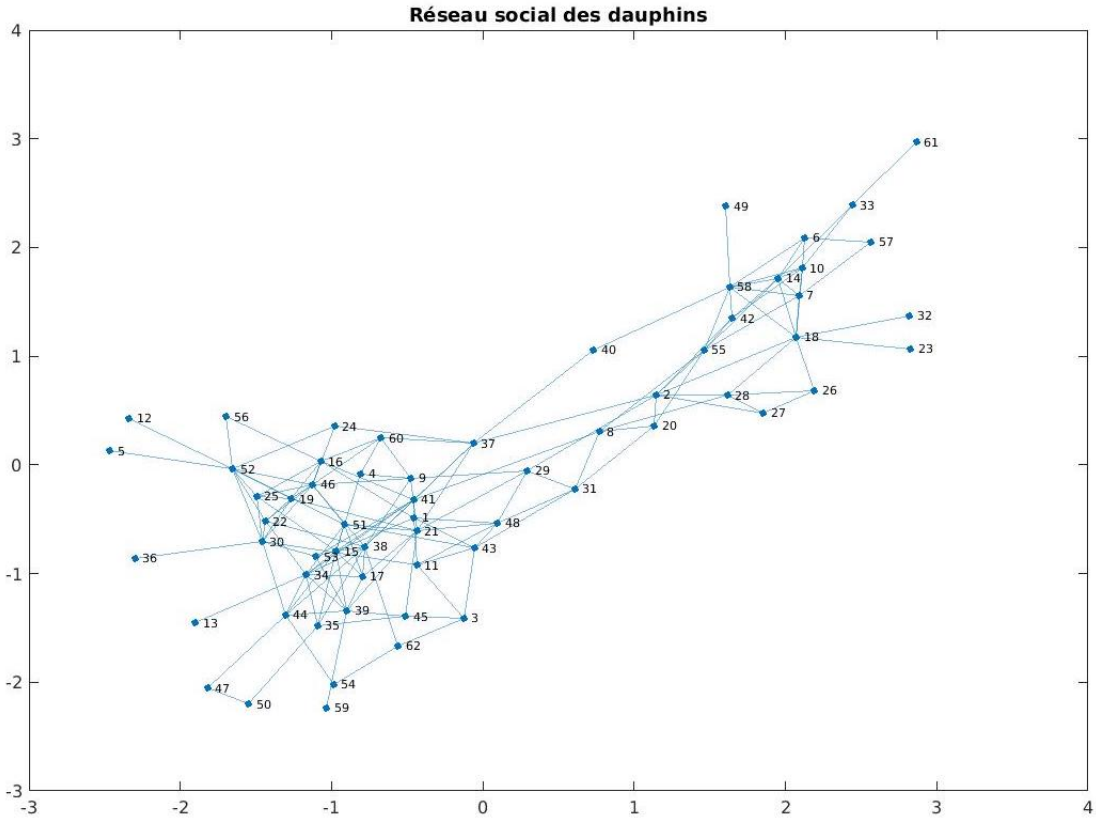

FIGURE A.4: Dauphins

Les misérables

Ce réseau décrit les apparitions au sein d'un même chapitre des personnages du livre « Les misérables » de Victor Hugo. Il possède les caractéristiques suivantes,

- ⋆ 77 sommets ;
- ⋆ 254 arêtes ;
- ⋆ non pondéré ;
- ⋆ non dirigé.

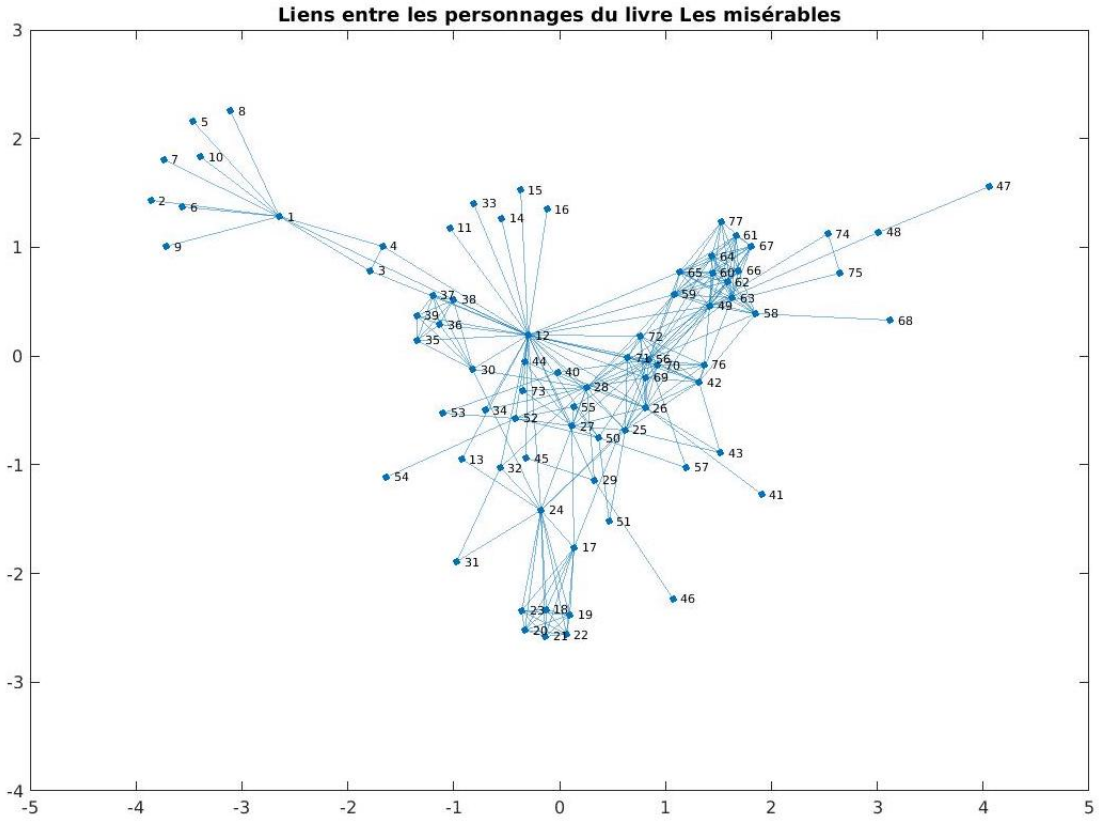

FIGURE A.5: Les misérables

David Copperfield

Les sommets de ce réseau représentent les noms et adjectifs du livre « David Copperfield » de Charles Dickens. Les arêtes quant à elles sont présentes lorsque ces éléments du langage sont utilisés de concert dans ce livre. Il possède les caractéristiques suivantes,

- ⋆ 112 sommets ;
- ⋆ 425 arêtes ;
- ⋆ non pondéré ;
- ⋆ non dirigé.

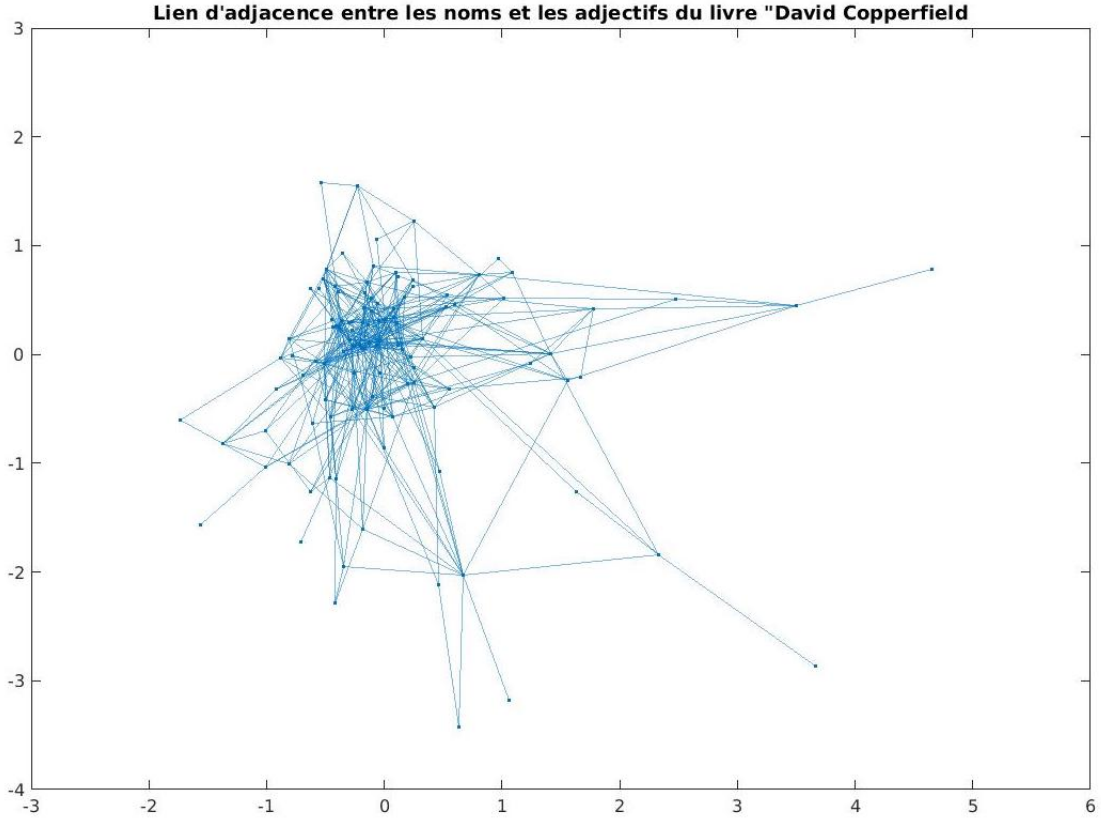

FIGURE A.6: David Copperfield

Annexe B

Implémentation de la méthode spectrale liée à l'inférence statistique

Matrice d'adjacence en entrée

```
function [stab, com, temps, j] = inference_A(A,T)

% [stab, com, temps, j] = inference(G,T) identifie la partition
% optimale du graphe associe a la matrice d'adjacence A pour
% le parametre de resolution T. T peut etre un vecteur ou non.
% Les sorties stab, com, temps, j representent respectivement
% la modularite generalisee, le nombre de communautes, le parametre
% de resolution et le nombre d'iterations associes a la repartition
% optimale pour chaque composants de T.

diff = 1;
j=1;
[x,y] = find(A);

while diff > 10^-6 && j <= 100
    [S, N, VI, C] = stability(A,T,'noVI','v');

    deg = 0.*[1:max(C)+1];
    min = 0;
    mout = 0;

    for i = 1:length(x)

        com1 = C(x(i))+1;
        com2 = C(y(i))+1;

        if com1 == com2
            min = min+1;
            deg(com1) = deg(com1)+2;
        else
            mout = mout+1;
            deg(com1) = deg(com1)+1;
```

```
            deg(com2) = deg(com2)+1;

        end

    end

    somme = 0;
    for i = 1:length(deg)

        somme = somme + deg(i)^2;

    end

    win = (2*min)/(somme/(2*length(A)));

    wout = (2*length(A) - min)/(2*length(A) - somme/(2*length(A)));

    gamma = (win - wout)/(log(win) - log(wout));

    temps(j) = T;
    stab(j) = S;
    com(j) = N;

    T = 1/gamma;
    if j==1
        diff = S;
    else
        diff = abs(S-S_old);
    end
    S_old = S;
    j = j+1;
end

end
```

Graphe en entrée

```
function [stab, com, temps, j] = inference(G,T)

% [stab, com, temps, j] = inference(G,T) identifie la partition
% optimale du graphe G pour le parametre de resolution T. G doit
% etre une liste des aretes du reseau definie comme suit, [sommet i,
% sommet j, poids de l'arete; sommet k, sommet l, poids de l'arete;
% ...]. T peut etre un vecteur ou non. Les sorties stab, com, temps, j
% representent respectivement la modularite generalisee, le nombre de
% communautes, le parametre de resolution et le nombre d'iterations
% associes a la repartition optimale pour chaque composants de T.


diff = 1;
j=1;

while diff > 10^-6 && j <= 100

    [S, N, VI, C] = stability(G,T,'noVI','v');

    deg = 0.*[1:max(C)+1];
    min = 0;
    mout = 0;
    for i= 1:length(G)

        com1 = C(G(i,1)+1)+1;
        com2 = C(G(i,2)+1)+1;

        if com1 == com2
            min = min+1;
            deg(com1) = deg(com1)+2;
        else
            mout = mout+1;
            deg(com1) = deg(com1)+1;
            deg(com2) = deg(com2)+1;
        end

    end

    somme = 0;
    for i = 1:length(deg)
```

```
        somme = somme + deg(i)^2;

    end

    win = (2*min)/(somme/length(G));

    wout = (length(G) - min)/(length(G) - somme/length(G));

    gamma = (win - wout)/(log(win) - log(wout));

    temps(j) = T;
    stab(j) = S;
    com(j) = N;

    T = 1/gamma;
    if j==1
        diff = S;
    else
        diff = abs(S-S_old);
    end
    S_old = S;
    j = j+1;

end

end
```

Annexe C

Variation d'information

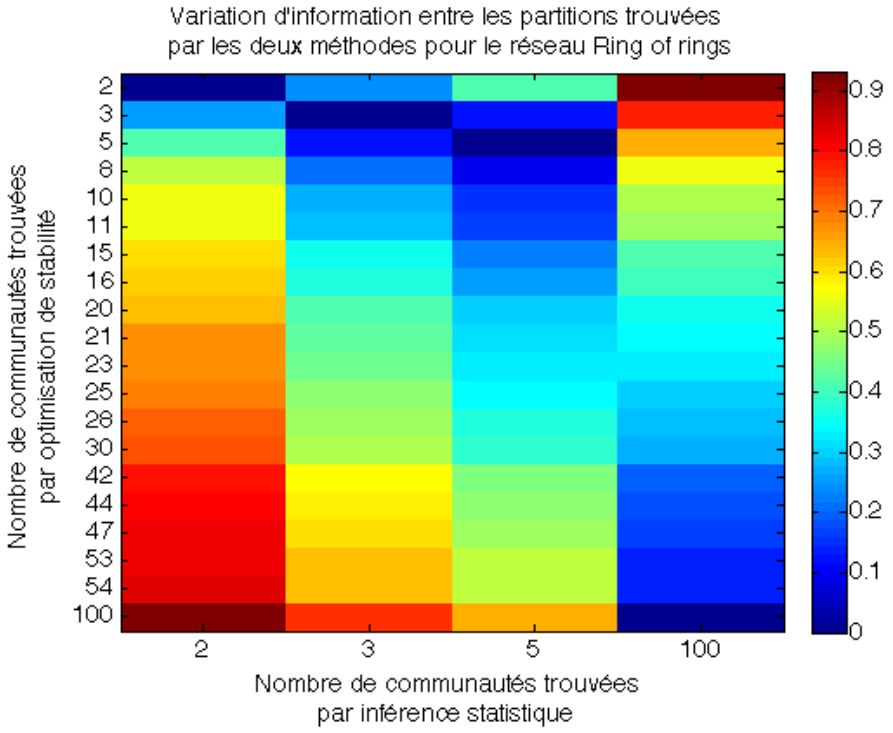

FIGURE C.1: Variation d'information entre les partitions obtenues par les deux méthodes pour le réseau « Ring of rings ».

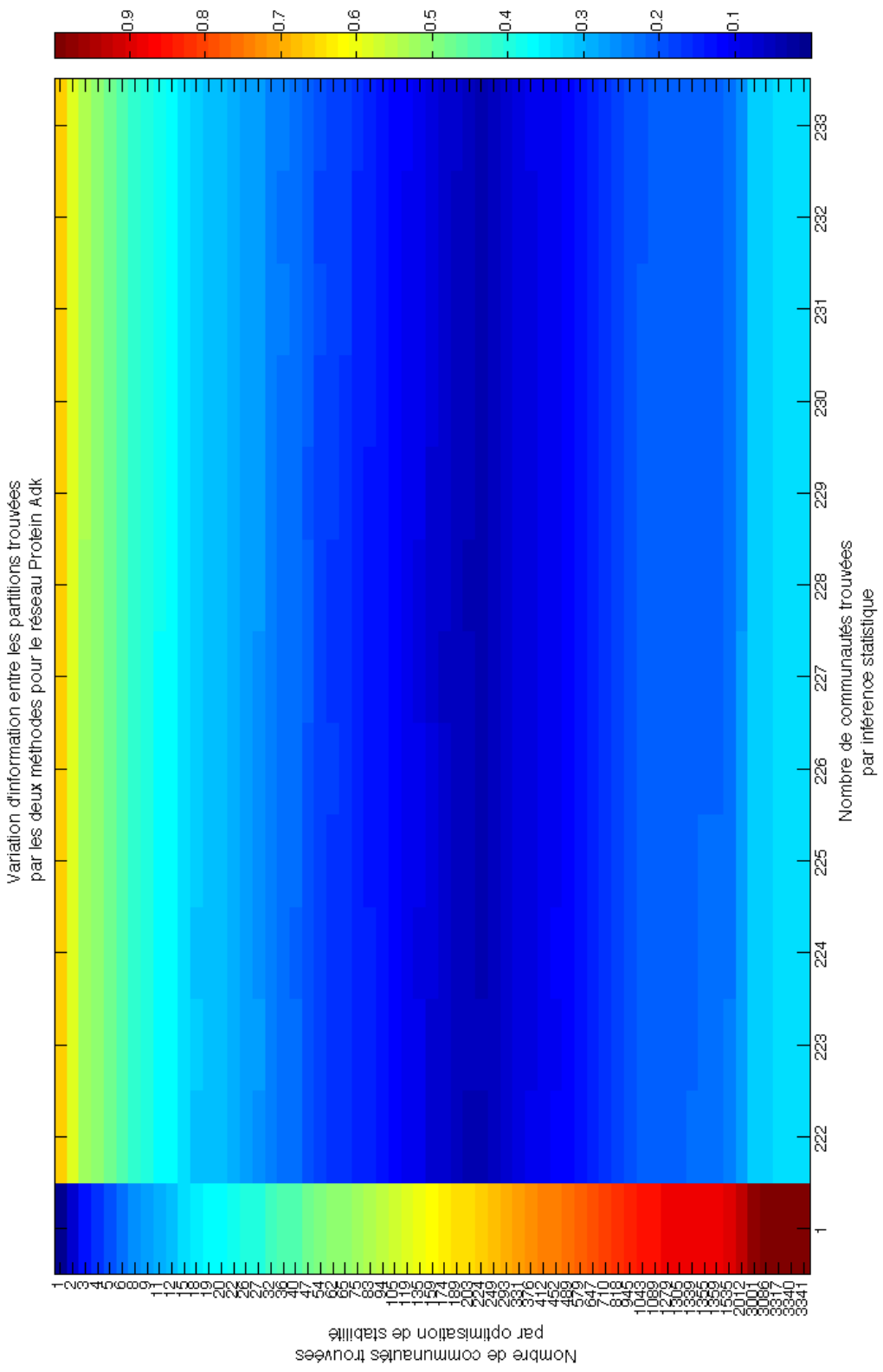

FIGURE C.2: Variation d'information entre les partitions obtenues par les deux méthodes pour le réseau d'une protéine Adk.

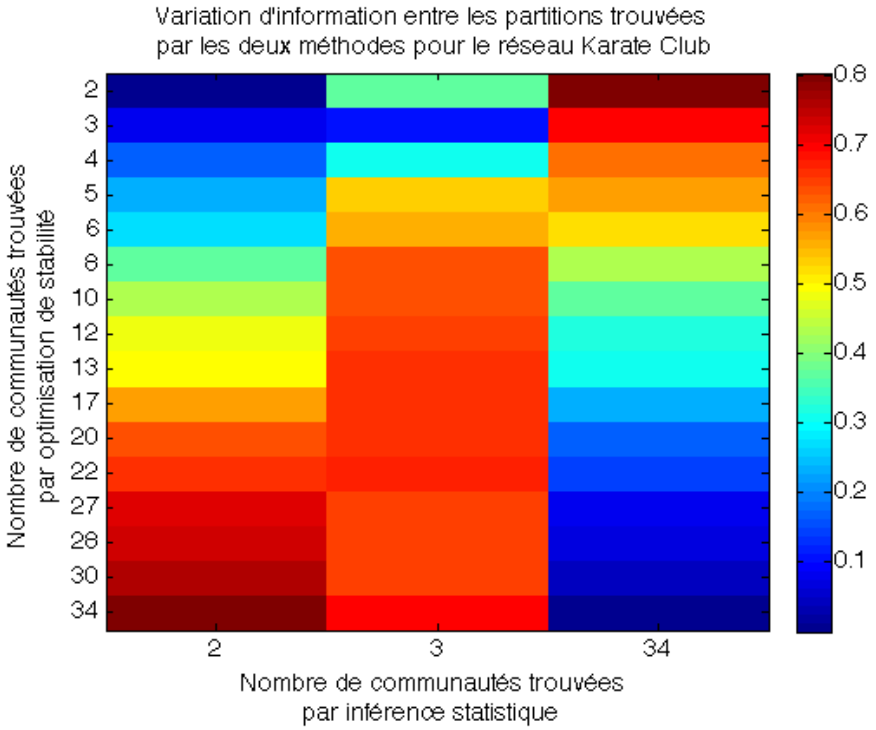

FIGURE C.3: Variation d'information entre les partitions obtenues par les deux méthodes pour le réseau du club de karaté.

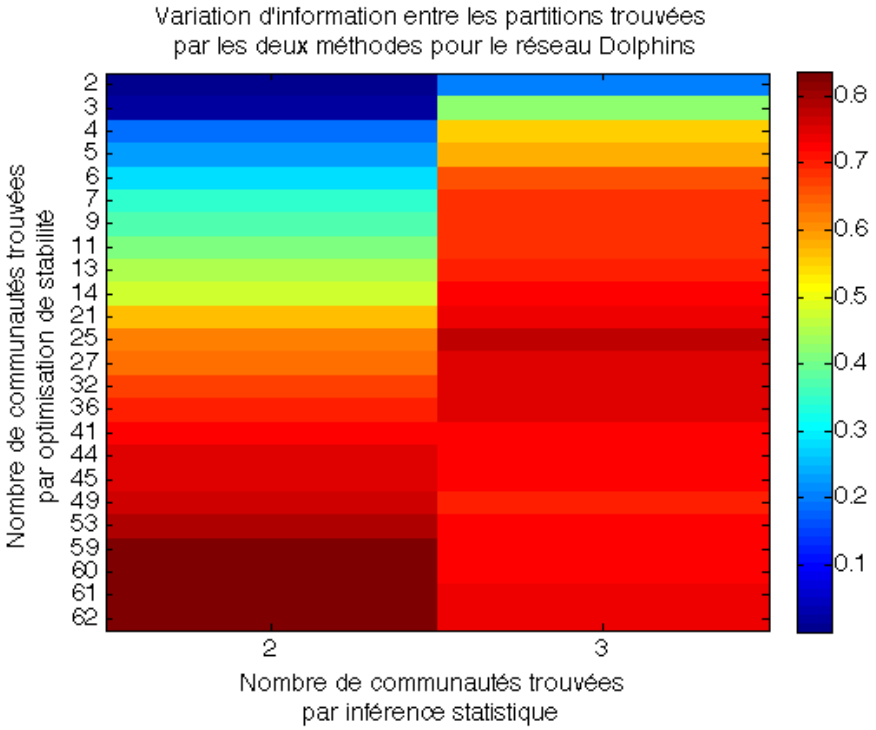

FIGURE C.4: Variation d'information entre les partitions obtenues par les deux méthodes pour le réseau d'une communauté de dauphins.

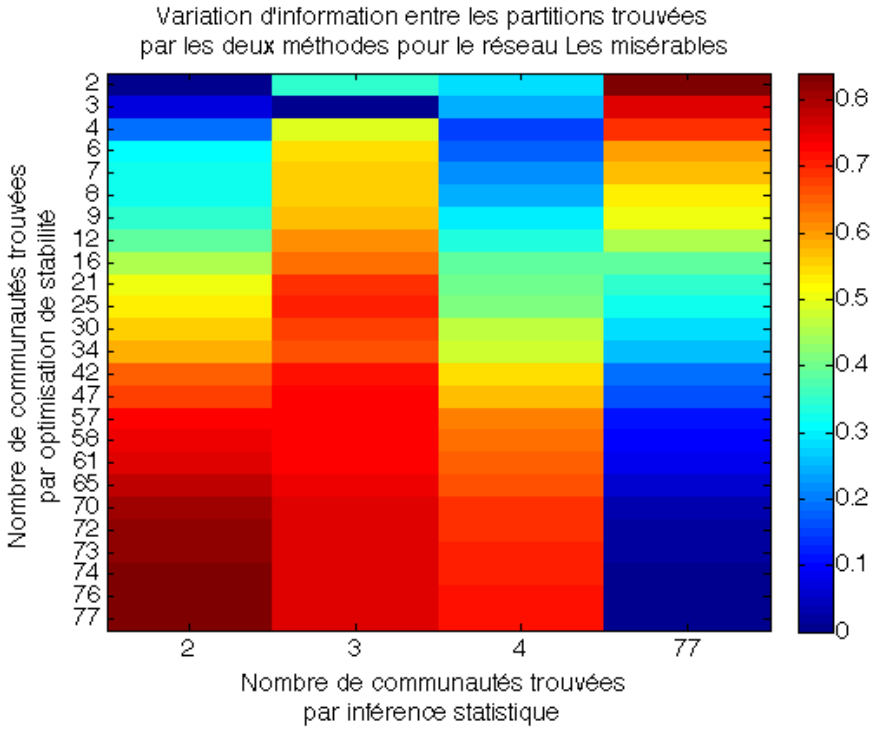

FIGURE C.5: Variation d'information entre les partitions obtenues par les deux méthodes pour le réseau des personnages du livre « Les misérables ».

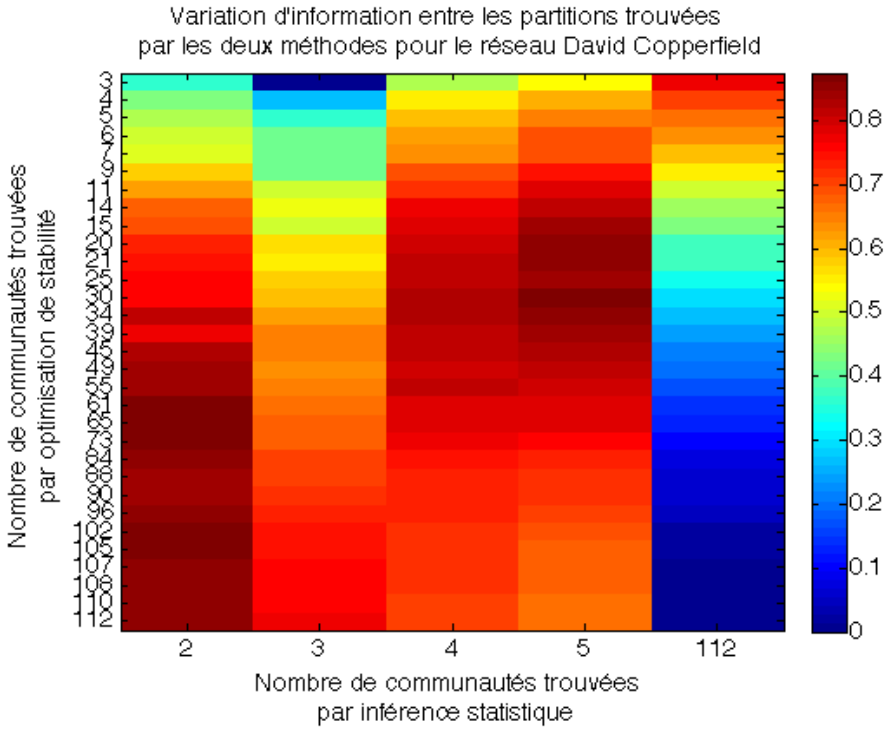

FIGURE C.6: Variation d'information entre les partitions obtenues par les deux méthodes pour le réseau des associations entre noms et adjectifs du livre « David Copperfield ».